FG

AD

ADA027670

R-TR-76-019

HYPERVELOCITY INFLIGHT TRAJECTORY SCATTER (HITS) CN CODE

USER'S MANUAL

ROBERT G. BELLAIRE
THEODORE O. GUSTAFSON

APRIL 1976

FINAL REPORT

PREPARED BY

AVCO CORPORATION
201 LOWELL STREET
WILMINGTON, MA 01887

CONTRACT NO.
DAAA09-74-C-2026-P00004

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

PREPARED FOR

RESEARCH DIRECTORATE
GENERAL THOMAS J. RODMAN LABORATORY
ROCK ISLAND ARSENAL
ROCK ISLAND, ILLINOIS 61201

HYPERVELOCITY INFLIGHT TRAJECTORY SCATTER (HITS) CODE

USER'S MANUAL

Prepared by

AVCO SYSTEMS DIVISION
201 Lowell Street
Wilmington, Massachusetts 01887

AVSD-0137-76-CR

Contract DAAA09-74-C-2026-P00004

Approved

Wm. G. Reinecke
Program Manager

Prepared for

GENERAL THOMAS J. RODMAN LABORATORY
Research Directorate
Rock Island Arsenal
Rock Island, Illinois 61201

April 1976

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: R-CR-76-019

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER: 9

4. TITLE (and Subtitle): 6 Hypervelocity Inflight Trajectory Scatter (HITS) Code. User's Manual.

5. TYPE OF REPORT & PERIOD COVERED: Final Report. Dec [redacted] 75 - Apr [redacted] 76

6. PERFORMING ORG. REPORT NUMBER: 14 AVSD-0137-76-CR

7. AUTHOR(s): 10 Robert G. Bellaire, Theodore O. Gustafson

8. CONTRACT OR GRANT NUMBER(s): 15 DAAA09-74-C-2026-P00004

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Avco Systems Division
201 Lowell Street
Wilmington, Massachusetts 01887

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62603-1W662603AH78-04-04

11. CONTROLLING OFFICE NAME AND ADDRESS:
Research Directorate, General Thomas J. Rodman Laboratory, Rock Island Arsenal, Rock Island, Illinois 61201

12. REPORT DATE: 11 Apr [redacted] 76

13. NUMBER OF PAGES: 378

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 12 368p.

15. SECURITY CLASS. (of this report): None

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report): Unlimited 16 DA-1-W-662603-AH-78

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report): 17 1-W-662603-AH-7804

18. SUPPLEMENTARY NOTES: 18 RIA-R 19 TR-76-019

19. KEY WORDS (Continue on reverse side if necessary and identify by block number):
dispersion
computer code user's manual
increased velocity projectile
exterior ballistics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number):

This report is a User's Manual for the HITS computer code. The HITS code statistically evaluates projectile dispersion, which is the dispersion associated with the inflight behavior of a gun launched projectile. Projectile dispersion is a fundamental limit on overall weapon system effectiveness. The code is an automated computer-based analytic procedure for computing crossrange and downrange dispersion error budgets and sensitivity → next page

LB

20. Abstract (Concl'd)

coefficients to the sources of projectile dispersion. The error budgets statistically define projectile dispersion at a level conducive to interpretation and comprehensive understanding.

HITS evaluates projectile dispersion by either analytic or Monte Carlo methods employing trajectory equations. The trajectory equations are closed form approximations to the six degrees of freedom equations of motion. The closed form equations presume the projectile does not experience transonic flow conditions during any phase of flight. This limits the present HITS code to this type of projectile (referred to here as "hypervelocity", with no other connotation intended). The trajectory equations have been subjected to extensive testing. In all cases they were accurate to within engineering tolerances. These equations have potential application to operational fire control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

Since this report is a User's Manual, the emphasis is placed on providing the information necessary to operate the code and interpret the results. Detailed input/output information is summarized in tables featuring step-by-step cook-book instructions. Example problems are discussed. Appendices present the theoretical and programming fine points, as well as the complete program listing.

FOREWORD

This User's Manual was prepared by Avco Systems Division, 201 Lowell Street, Wilmington, Massachusetts 01887, for the U. S. Army Armament Command, Gen. T. J. Rodman Laboratory, Rock Island Arsenal, Rock Island Illinois 61201, with Mr. William P. Wohlford of the Research Directorate as contract monitor.

The principal authors appreciate the contributions and suggestions offered by their colleagues: P. Crimi, R. Flaherty, P. Fote, R. Kingsley, W. Reinecke, and R. Torrani.

The authors wish to thank the many Rock Island personnel, and particularly Mr. William P. Wohlford and Mr. Charles S. Hicks, who provided Avco with much timely and relevant data and many sound suggestions.

The authors also wish to thank Mrs. L. Kiley, Mrs. J. Di Pietro, Mrs. P. Ford and Miss M. Smith for their careful and conscientious typing, Mrs. H. Cassell for the many equations, Miss A. Gaudet and Mr. R. Gaumond for their excellent art work, and Mr. E. Schurmann and Mr. L. Sacks for their editorial assistance.

ABSTRACT

This report is a User's Manual for the HITS computer code. The HITS code statistically evaluates "projectile dispersion," which is the dispersion associated with the inflight behavior of a gun launched projectile. Projectile dispersion is a fundamental limit on overall weapon system effectiveness. The code is an automated computer-based analytic procedure for computing crossrange and downrange dispersion error budgets and sensitivity coefficients to the sources of projectile dispersion. The error budgets statistically define projectile dispersion at a level conducive to interpretation and comprehensive understanding.

HITS evaluates projectile dispersion by either analytic or Monte Carlo methods employing trajectory equations. The trajectory equations are closed form approximations to the six degrees of freedom equations of motion. The closed form equations presume the projectile does not experience transonic flow conditions during any phase of flight. This limits the present HITS code to this type of projectile (referred to here as "hypervelocity", with no other connotation intended). The trajectory equations have been subjected to extensive testing. In all cases they were accurate to within engineering tolerances. These equations have potential application to operational fire control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

Since this report is a User's Manual, the emphasis is placed on providing the information necessary to operate the code and interpret the results. Detailed input/output information is summarized in tables featuring step-by-step cook-book instructions. Example problems are discussed. Appendices present the theoretical and programming fine points, as well as the complete program listing.

TABLE OF CONTENTS

PRECEDING PAGE BLANK-NOT FILMED

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Concl'd

Page

LIST OF FIGURES

Figure		Page

LIST OF FIGURES (CONT'D)

LIST OF FIGURES (CONCL'D)

LIST OF TABLES

1.0 ENGINEERING PERSPECTIVE

The Hypervelocity[1] Inflight Trajectory Scatter (HITS) computer code is an automated procedure for evaluating "projectile dispersion", which is the dispersion associated with the inflight behavior of a gun launched projectile. This report documents the code in the format of a User's Manual. Primary emphasis is placed on the information necessary to operate the code and interpret the results. Theoretical aspects and computational details are treated in the appendices.

This chapter addresses the following. Section 1.1 identifies the sources of projectile dispersion and, thereby, more clearly defines the term. Section 1.2 discusses the relationship of projectile dispersion to overall weapon system effectiveness. The dispersion analysis methodology suggested by the code is illustrated by example in Section 1.3. Section 1.4 summarizes the limitations of the code. An overview of the report is presented in Section 1.5. The overall objective of this chapter is to enable the engineer to quickly determine the utility of HITS with respect to his particular problem.

1.1 Projectile Dispersion Sources

The HITS computer code quantifies projectile dispersion. Projectile dispersion is defined here to be the dispersion associated with inflight behavior which cannot be compensated by fire control. (HITS does not compute the dispersion attributable to fire control. However, it does include in the projectile dispersion calculation the effects of fire control corrections for inflight phenomena.) Thus, the sources of projectile dispersion enumerated in this section are those factors which influence inflight behavior and cause the fire control predicted trajectory to differ from the true flight path.

Table 1-1 lists the sources of projectile dispersion. The left hand column contains the physical mechanisms through which the contributing factors of the right hand column act. There are four general categories: (1) initial conditions at the muzzle, (2) projectile mass properties, (3) projectile aerodynamic characteristics, and (4) atmospheric effects. The sources of Table 1-1 cause dispersion only insofar as the fire control computer cannot anticipate their effects and apply corrective action. For instance, winds are a source of projectile dispersion if the weapon system does not measure them. If it

[1] "Hypervelocity" is used throughout this report to indicate the limitation of the present HITS code to projectiles whose velocity does not become transonic at any point along the trajectory. No other connotation is intended.

Table 1-1 Sources of Projectile Dispersion

ERROR SOURCE	TYPICAL CONTRIBUTING FACTORS
Initial Conditions: Velocity Vector Angle of Attack Angular Rates	Sabot Separation In-Barrel Dynamics Blast Effects Barrel Vibrations Tipoff Rates Gun Slewing Rates
Inertial Characteristics: Weight Moments of Inertia Physical Dimensions C. G. Asymmetries	Manufacturing Tolerances
Aerodynamic Characteristics: Static Coefficients Dynamic Coefficients Spin Rate Trim Angle of Attack Static Margin	Manufacturing Tolerances Ablations Effect on the Ballistic Coefficient Coefficient Measurement Errors
Atmospheric Effects: Winds Density	Temporal and Spatial Atmospheric Variations

does, only the wind measurement error causes projectile dispersion. Although not mentioned in Table 1-1, simplifying assumptions incorporated in the fire control trajectory to minimize computational requirements are also considered sources of projectile dispersion. The HITS code accounts for each of these factors.

1.2 Weapon System Effectiveness Implications

This section discusses the role projectile dispersion plays in determining weapon system effectiveness with implications to projectile design procedures. The discussion opens with a brief overview of the factors which affect weapon system effectiveness.

Figure 1-1 is a block diagram of a modern gun weapon system. It indicates the various factors that influence weapon system effectiveness, as defined by the probability of kill. Referring to Figure 1-1, the engagement scenario shown on the left hand side establishes the geometry of the encounter. Target motion dynamics limit target motion to physically possible rates and accelerations. The acquisition sensor continually measures the target coordinates and passes the information to the fire control computer. Fire control's basic jobs are to solve the intercept geometry, align the gun, and signal the gunner at the appropriate time. Its calculations take into account not only the target coordinates but also information concerning the inflight behavior of the projectile, winds, atmospheric density, gun orientation, and various system models describing target dynamic constraints, gun mount rate and orientation limits, etc. Fire control issues commands to the gun mount servo system to point the gun. The servo holds the gun on the target while awaiting the command to fire. After the round has been fired, the projectile trajectory relative to the true target position determines the point of closest approach, that is the miss distance. If a hit is scored, the location of the hit and the projectile terminal ballistics determine the probability of kill.

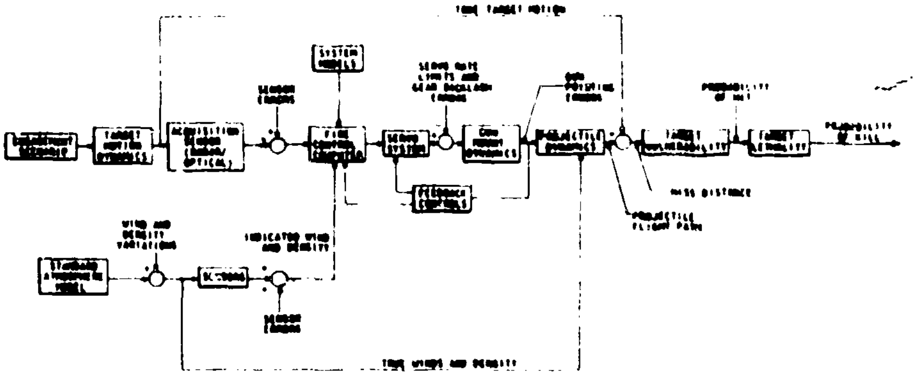


Figure 1-1 Weapon System Effectiveness

Projectile dispersion affects overall weapon system effectiveness. The accuracy with which the fire control computer can predict the true flight path of the projectile poses a fundamental limitation on weapon system effectiveness. Even if the exact position of the target relative to the gun were known and the target could be tracked with infinite precision, the inability of fire control to predict the flight path exactly would cause the projectile to miss the target. Thus, projectile dispersion is a fundamental limit on weapon system effectiveness because it defines a level of accuracy which cannot be improved upon by refinements of other elements of the weapon system. The HITS code quantitatively defines this limit.

HITS may also be used as a projectile design tool. Since all projectiles exhibit dispersion to a greater or lesser extent and this can limit overall weapon system effectiveness, it would be prudent to include the evaluation of projectile dispersion in all projectile design studies. The HITS code is an analytic tool for performing this evaluation inexpensively. Furthermore, HITS can compute a single trajectory under a given set of conditions or under a cyclic permutation of parameters. The calculation is quick, accurate, and inexpensive. Thus, HITS brings to design activities, additional capabilities that go beyond the assessment of projectile dispersion.

Although projectile dispersion poses a fundamental limit on weapon system effectiveness, it is not always clear what impact projectile dispersion will have on overall weapon system effectiveness. Returning to Figure 1-1, it is clear that overall weapon system effectiveness ultimately depends on how well the gun system operates as a whole. There is a synergistic effect which makes the system better than the sum of its parts, since one subsystem can be designed to compensate for the errors of another. For instance, fire control could be designed to monitor and correct for gun pointing errors, as indicated by the feedback loop shown in Figure 1-1. Thus, fire control can point the gun better than an analysis of the gun-mount servo system would indicate was possible. For this reason, it is potentially misleading to forecast overall weapon system effectiveness from an analysis of any single subsystem (e.g., a projectile dispersion analysis). Thus, judgment must be used in interpreting projectile dispersion assessments produced by the HITS code in terms of overall weapon system effectiveness.

1.3 Dispersion Analysis Methodology

The HITS computer code suggests a methodology for determining the susceptibility of projectile designs to projectile dispersion. This section demonstrates the two step method with an illustrative example. The following paragraphs illustrate the information required, the role of the HITS code, and the format of the results.

Table 1-1 lists the sources of projectile dispersion. By analyzing the contributing factors, statistics may be assigned to each source which quantify the uncertainty as illustrated in Table 1-2: the error source model. Displayed in the left hand column of Table 1-2 are the sources of projectile dispersion. The third column contains the nominal values for the projectile properties. The second column defines the distribution of variations about the nominal. The uncertainties are given in the fourth column: the standard deviations or one-sigma (1-σ) values.[1] The standard deviations quantify the level of uncertainty in the information supplied to fire control. The error source model may be even more detailed. The basic error source model of Table 1-2 could be expanded to include correlations between the elements of the error source model, such as projectile weight and muzzle velocity. Whatever the level of detail, construction of the error source model concludes the first step in evaluating projectile dispersion via the HITS code.

The second and last step is to use the HITS code to evaluate the projectile dispersion corresponding to the error source model as suggested by the schematic block diagram of Figure 1-2. The code reads the error source model as data and proceeds to quantify projectile dispersion in the form of an error budget. The code contains two sets of closed form trajectory equations: one represents the true trajectory and the other represents fire control. A Statistical Processor manipulates the trajectory equations in either an analytical or Monte Carlo fashion to determine the dispersion statistics. The results of the calculations are most conveniently summarized in the format of a projectile dispersion error budget as indicated in Figure 1-2. Any simplifying assumptions incorporated in an actual fire control trajectory are input to the code and taken into account.

The HITS code can construct error budgets in both crossrange and downrange directions as well as evaluate three dimensional dispersion indices such as the Spherical Error Probable (SEP). Error budgets may be constructed at either nominal time or nominal range. In this respect HITS is very flexible and can compute most every statistic customarily used to describe dispersion.

---

1 Rayleigh/Uniform uncertainties are specified by the mean magnitude given in the third column.

Table 1-2 Error Source Model

ERROR SOURCE	STATISTICAL DISTRIBUTION	MEAN VALUE	STANDARD DEVIATION
● INITIAL CONDITIONS			
● VELOCITY VECTOR			
- MAGNITUDE	Gaussian	11,000 ft/sec	1/3%
- ORIENTATION	Rayleigh/Uniform*	0.0005 deg	--
● INERTIAL ORIENTATION			
- ATTITUDE	Rayleigh/Uniform*	0.1 deg	--
- ATTITUDE RATE	Rayleigh/Uniform*	55 rad/sec	--
● PHYSICAL CHARACTERISTICS			
● WEIGHT	Gaussian	0.11 lbs	1.0%
● MOMENTS OF INERTIA			
- AXIAL	Gaussian	1.234×10^{-6} slug-ft^2	1 2/3%
- PITCH	Gaussian	2.110×10^{-5} slug-ft^2	1 2/3%
● PHYSICAL DIMENSIONS			
- REFERENCE AREA	Uniform	3.068×10^{-3} ft^2	2/3%
- LENGTH	Uniform	0.31 ft	1/3%
- BASE DIAMETER	Uniform	0.0625 ft	1/3%
● AERODYNAMIC CHARACTERISTICS			
● STATIC COEFFICIENTS			
- DRAG VARIATION EFFECTS			
VELOCITY	Gaussian	**	1.0%
ANGLE OF ATTACK	Gaussian	1.0 1/deg^2	1.0%
- NORMAL FORCE	Gaussian	1.9767 1/rad	2.0%
● DYNAMIC COEFFICIENTS			
- PITCH DAMPING	Gaussian	-7.5 (none)	20.0%
- MAGNUS MOMENT	Gaussian	0.0 (none)	0.0
● SPIN RATE	Gaussian	400 rad/sec	1.0%
● TRIM ANGLE OF ATTACK	Rayleigh/Uniform*	0.1 deg	--
● STATIC MARGIN	Gaussian	6.2% length	1.0% of length
● ATMOSPHERIC EFFECTS			
● CONSTANT WINDS	Rayleigh/Uniform*	11.0 ft/sec	--
● DENSITY VARIATIONS	Gaussian	2.378×10^{-3} slug-ft^3	1.0%

*Denotes a Rayleigh distribution of magnitude with a Uniform 360° distribution in orientation.

**Drag variation with velocity closely approximated by $C_{X_o} = C_{X_\infty} + \frac{K_D}{V^2}$ which is fit to two data points $(C_{X_1}, V_1) = (0.03585, 16{,}740)$ and $(C_{X_2}, V_2) = (0.11951, 3906)$. Uncertainty in C_{X_1} and C_{X_2}.

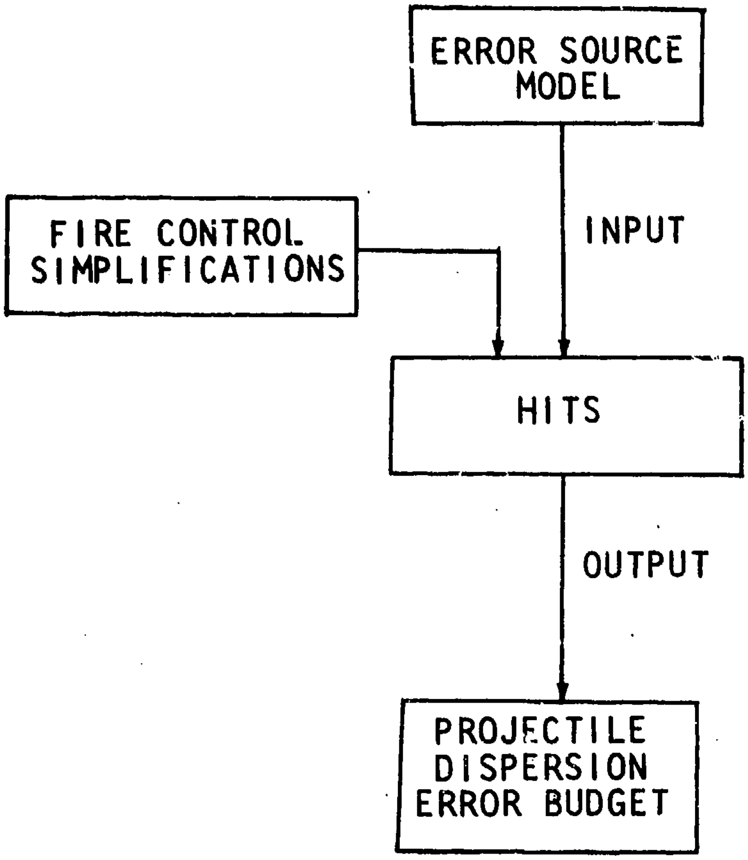


Figure 1-2 HITS Overview

Table 1-3 presents the error budget generated by HITS for crossrange dispersion at nominal time corresponding to the error source model of Table 1-2. The error budget presents both the dispersion and the sensitivity coefficient associated with each error source. This identifies the larger contributors and those with significant potential. The total dispersion is given in the lower right hand corner. Table 1-3 illustrates the level of detail and types of information available from HITS. This is precisely the level of detail the engineer needs to assess the situation and make decisions. Supplemental information concerning the dispersion probability distribution can be developed if desired.

In summary, the methodology suggested by the HITS code consists of (1) constructing a comprehensive error source model and (2) processing it to obtain a detailed error budget. The error budget contains the information necessary to evaluate a projectile with respect to projectile dispersion program objectives and/or trade-off rival candidate designs. The method could also be employed in conjunction with ballistic range tests devoted to dispersion assessment with the objective of deriving additional insight. Whatever the motive, HITS provides a scientific, systematic method of evaluating projectile dispersion.

Presentation of the error source model and error budget, Tables 1-2 and 1-3, requires comment concerning their generality. The error source model contains implicit assumptions concerning (1) manufacturing tolerances on the projectile and barrel, (2) the amount of testing employed to determine the aerodynamic characteristics, and (3) the presumed accuracy of wind and density measurements. The error budget reflects these assumptions and further assumes the fire control trajectory algorithm contains no simplifying approximations. Thus, the error source model and error budget presented here are only for the purposes of illustration. They are not necessarily representative of any weapon system in the inventory, under procurement, or in development.

1.4 Code Limitations

This section states the limitations of the HITS code. Most of the items listed below are not theoretical limitations. Some are assumptions made for modeling convenience, while others are merely computer requirements. In any case, they are listed

Table 1-3 Crossrange Dispersion at Nominal Time
(Nominal Range = 10 Kft)

ERROR SOURCE	UNCERTAINTY	SENSITIVITY (MRAD/·)	1-σ ERROR (MRAD)
• INITIAL CONDITIONS			
• VELOCITY VECTOR			
- MAGNITUDE	1/3%	0.001913	0.0006377
- ORIENTATION	0.0005 deg	13.91	0.006957
• INITIAL ORIENTATION			
- ATTITUDE	0.1 deg	0.01551	0.001551
- ATTITUDE RATE	55.0 rad/sec	0.02346	1.290
• PHYSICAL CHARACTERISTICS			
• WEIGHT	1.0%	0.009808	0.009808
• MOMENTS OF INERTIA			
- AXIAL	1 2/3%	≈0	≈0
- PITCH	1 2/3%	≈0	≈0
• PHYSICAL DIMENSIONS			
- REFERENCE AREA	2/3%	0.003439	0.002292
- LENGTH	1/3%	0.01274	0.004248
- BASE DIAMETER	1/3%	0.00002225	0.000007418
• AERODYNAMIC CHARACTERISTICS			
• STATIC COEFFICIENTS			
- DRAG VARIATION EFFECTS			
VELOCITY	1.0%	≈0	≈0
ANGLE OF ATTACK	1.0%	0.000007418	0.000007418
- NORMAL FORCE	2.0%	0.0008093	0.001619
• DYNAMIC COEFFICIENTS			
- PITCH DAMPING	20.0%	0.00001131	0.0002261
- MAGNUS MOMENT	0.0%	--	--
• SPIN RATE	1.0%	0.0007241	0.0007241
• TRIM ANGLE OF ATTACK	0.1 deg	0.8103	0.08103
• STATIC MARGIN	1% L	0.2183	0.2183
• ATMOSPHERIC EFFECTS			
• CONSTANT WINDS	11 ft/sec	0.01538	0.1692
• DENSITY VARIATIONS	1.0%	0.003429	0.003429
		RSS TOTAL DISPERSION	1.322

below to aid the engineer in determining whether HITS is applicable to a particular problem and, if so, facilitate installation. The limitations are:

- The projectile drag model assumes the projectile does not experience transonic flow conditions at any time from launch to impact. A transonic condition would cause appreciable error only if it persisted over a significant portion of the flight time. This assumption can be overcome by additional code development. However, the code as presently configured is limited to "hypervelocity" projectiles.

- Muzzle blast, sabot separation and in-barrel dynamic effects are combined and represented by equivalent velocity and attitude perturbations at the muzzle. Ballistic range data reduction methods typically combine these effects in this manner.

- Wind gusts and density variations, that is fluctuations along the trajectory, are assumed to be zero. Covariance analysis using available wind gust spectra suggest wind gusts are a negligible source of dispersion for typical projectiles.[1] On the other hand, steady winds are known to be a significant source and are accounted for by the HITS code.

- When operated in the Monte Carlo mode, HITS cannot assess the effects of error source model correlations. A program modification would be required. Correlation effects may, however, be evaluated via the Analytical Statistical mode.

- The HITS code is written in FORTRAN IV and requires approximately 175K bytes of computer memory. The card deck is punched in the EBCDIC format.

[1] Gustafson, T. G., Crimi, P., Bellaire, R. G., "Dispersion of Increased Velocity Projectiles - Feasibility Phase," Avco Corporation, AVSD-0200-75-CR, July 1975.

- The random number generator (Subroutine RANDU) will work properly only on a 32 bit word computer, such as an IBM-360. The subroutine can be easily modified. However, some care must be exercised in performing the modification to insure the generated random sequences will reproduce those of the original code on a 32 bit word computer. Otherwise, the full instructional benefits of the example problems presented here will not be achieved, since it will not be possible to reproduce the Monte Carlo simulations.

- The code contains an interface for automatic computer plots. Since plotting software is hardware specific, this capability is not fully developed. The interface is thoroughly documented.

- The code was designed to facilitate real-time interactive mode operation from remote keyboard terminals. This capability is latent in the code, but not an operational reality.

1.5 Report Overview

Since this report is a User's Manual for the HITS code, the emphasis is on providing the information necessary to operate the code and interpret the results. The text presents the day-to-day necessities required to perform dispersion analyses. The appendices discuss the theoretical and programming aspects.

Chapter 2 presents an overview of the HITS code directed to the user. The objective is to aquaint the analyst quickly with the code. The basic code structure and options are described. The trajectory equations, which are the heart of the code, are discussed from a functional point of view. Chapter 2 concludes with a collection of topics peripheral to HITS, but pertinent to dispersion analysis.

Detailed input information is supplied in Chapter 3. A quick-reference step-by-step format is employed. Tables are used to define all variables and present all relevant information; comments direct the user to sections of this report containing related indepth discussions. Chapter 3 closes with an illustrative example demonstrating the encoding of the error source model of Table 1-2.

Chapter 4 presents four input-output example problems. Since the computer printed output is self-explanatory, the initial impression might be that these examples are superfluous. They are not. They illustrate the output format and provide numerical check problems. The accompanying text stresses interpretation. Thus, users are encouraged to reproduce these results to familiarize themselves with HITS and to verify the code at their facility.

A brief summary is presented in Chapter 5. The antecedants of the HITS code and related analyses are discussed.

The appendices relate the theoretical and programming aspects of the HITS code. Appendix A is a summary discussion of the relevant aspects of probability theory and statistics. Appendix B discusses programming fine points. Appendix C mathematically develops the trajectory equations. Appendix D contains the complete program listing.

2.0 OVERVIEW

The purpose of this chapter is to discuss the salient aspects of the HITS code. The objective is to present the user with a clear picture of the essence and function of the code as a whole. To this end, analytical and programming fine points are glossed over in an effort to place maximum emphasis on the big picture. Detailed "how to do it" information is presented in later chapters. Theoretical analyses and programming information are presented in the appendices.

Since projectile dispersion is most meaningfully stated in a statistical sense, the HITS code quantifies it in statistical terms. Thus, familiarity with the basic concepts of probability and statistics is essential to (1) the preparation of the engineering formulation of the HITS input, (2) a thorough understanding of the algorithms employed by HITS, and (3) the interpretation of the HITS output. Appendix A reviews the most pertinent elements of probability and statistics with the objective of supplying this understanding. HITS can be operated by personnel unfamiliar with probability and statistics. Given the information presented in Chapter 3, they can encode the formulated problem and obtain the desired numerical results.

Section 2.1 discusses the structure of the code. Analysis options are enumerated in Section 2.2. The close-form trajectory equations, which are the heart of the HITS code, are described in Section 2.3. Section 2.4 discusses three topics which are peripheral to HITS, but pertinent to dispersion analysis.

2.1 Code Structure

A flow diagram illustrating the basic structure of the HITS code is shown in Figure 2-1. As suggested by the figure, HITS basic function is to process an error source model to obtain a projectile dispersion error budget. This section describes the three computational modules that perform this task: (1) the Input Processor, (2) the Statistical Processor, and (3) the Projectile Trajectory Module. The code contains facilities for accessing a Time Phased Data Base for updating projectile characteristics under direction from a remote interactive computer terminal; however, this capability is not an operational reality at the present time. Sections 2.1.1 and 2.1.2 discuss the function of the Input and Statistical Processors which were taken from previously developed software. Section 2.1.3 functionally describes the Projectile Trajectory Module, which was developed specifically for dispersion assessment.

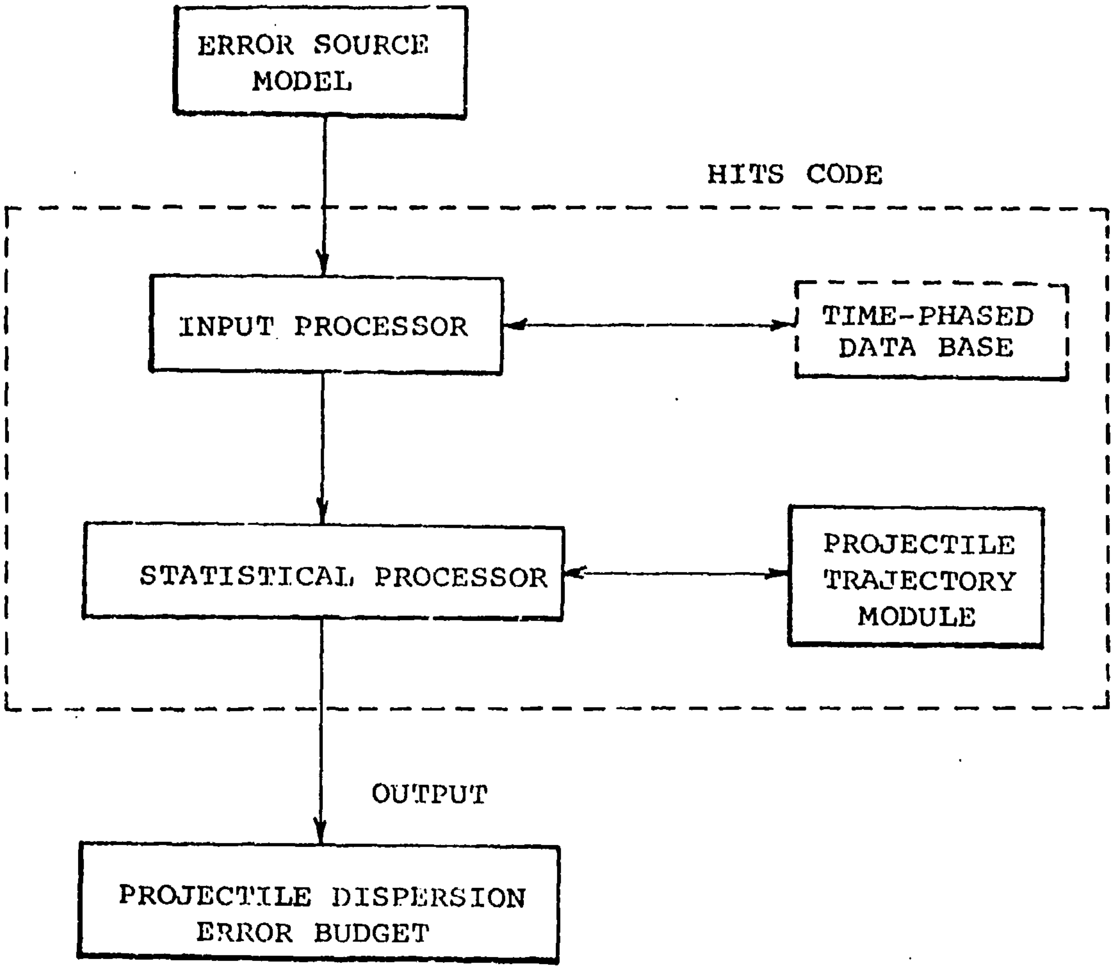


Figure 2-1 HITS Flow Diagram

2.1.1 Input Processor

The Input Processor reads error source model information from the input data stream. Two operations are performed once all the data has been read: (1) the input data is checked against a master list to determine whether crucial information is missing, and (2) the experimentor is informed as to the completeness of the stated error model. Missing data is automatically filled in with preset values.

The input data must define each element of the error source model (be it aerodynamic coefficient, physical dimension, or whatever) either as a constant or as a random variable. Random variables must be defined as to distribution type, mean or nominal value, and standard deviation (i.e., the 1-σ uncertainty). Gaussian, Rayleigh and uniform random variables are particularly easy to input, but any distribution is accepted. The input information is passed to the Statistical Processor.

2.1.2 Statistical Processor

The Statistical Processor derives dispersion statistics by referring to the Projectile Trajectory Module, as suggested by Figure 2-1. Each reference is the analytic equivalent of a ballistic range shot fired under the conditions specified by the Trajectory Module inputs. The Statistical Processor manipulates the Trajectory Module inputs in accord with the error source model uncertainties. The shot-by-shot dispersion returned by the Trajectory Module is compiled by the Statistical Processor to determine the dispersion error budget statistics.

2.1.3 Projectile Trajectory Module

The Projectile Trajectory Module consists of two sets of closed form parametric trajectory equations. The first computes the Fire Control predicted trajectory and the second simulates the true flight path or "Real World" Trajectory. The Trajectory Module returns the difference to the Statistical Processor.

The Fire Control equations are solved for the nominal time required to reach nominal range and the coordinates and velocity of the projectile at that time. This fixes the aim point in time as well as space. Typically, the parameters of the Fire Control equations are set equal to the error source model nominal values, so that the aim point is based on nominal muzzle velocity, weight, drag, etc. Simplifying assumptions found in operational

Fire Control computers can be simulated by judiciously selecting these parameters. Dispersion is computed relative to the aim point.

The Real World trajectory equations are solved for the actual projectile coordinates and velocity at nominal time and at nominal range. The distance between the projectile position at nominal time and the aim point defines dispersion for engagements with rapidly moving targets, where time of arrival is important. The displacement of the projectile position at nominal range from the aim point defines dispersion for engagements with slowly moving targets. The parameters of the Real World equations change from shot to shot and contain perturbations in accord with the error source model uncertainties. Thus, the dispersion returned to the Statistical Processor is consistent with the error source model and incorporates the effects of Fire Control simplifying assumptions.

2.2 Analysis Options

The Statistical Processor of Figure 2-1 was extracted from previously developed software. As a result, it has several desirable features unrelated to statistical dispersion assessment. The Statistical Processor has four modes of operation: (1) Single Trajectory, (2) Range Check, (3) Analytical Statistical, and (4) Monte Carlo. The first two modes are of primary interest when designing projectiles or analyzing various launch phenomena. The latter two modes are statistical methods for quantifying dispersion. The following subsections define the modes and suggest usages.

2.2.1 Single Trajectory

This mode computes the dispersion associated with a single shot at a given range. The Single Trajectory mode would be particularly useful in selecting nominal design parameters such as spin rate. A nominal trim lift would be input and the spin rate varied to minimize dispersion at maximum range.

2.2.2 Range Check

This mode evaluates the dispersion associated with a systematic variation of input parameters, thereby providing valuable design data. For instance, a Range Check on nominal range (5000, 6000, ---, 10,000 ft) would generate a complete trajectory. Alternatively, a Range Check on static margin (4, 5, and 6%) and spin rate (300, 400, and 500 rad/sec) would produce simulated shots with all possible combinations of static margin and spin rate. Comparison of the dispersions might suggest the optimal combination.

2.2.3 Analytical Statistical

The Analytical Statistical mode evaluates first and second order dispersion sensitivity coefficients by varying the trajectory inputs about the error source nominal. This mode computes, for example, the jump angle sensitivity to cross wind uncertainty; that is, the milliradians per ft/sec. The Analytical Statistical mode then uses the sensitivity coefficients to evaluate the average dispersion as well as the 1-σ uncertainty. The dispersion probability distribution is not determined. The analytic techniques employed permit rapid, low-cost dispersion assessment.

2.2.4 Monte Carlo

The Monte Carlo mode simulates ballistic range tests. It evaluates not only dispersion statistics (i.e., mean values and σ's) but also the dispersion distribution. Numerous trajectories are computed using computer generated random numbers to simulate error source model uncertainties. Each simulated trajectory is comparable to a ballistic range shot, at a fraction of the cost. A test sequence is summarized by statistical indices and bar charts showing the number of times the projectiles had dispersions of say 0 to 0.25 milliradians, 0.25 to 0.50 milliradians, etc. The Monte Carlo mode gives the most complete dispersion picture.

2.3 Trajectory Equations

The Projectile Trajectory Module contains both Fire Control and Real World closed form trajectory equations. Both mechanize the theory presented in Appendix C. Section 2.3.1 discusses the rationale for the selection of closed form equations. An overview of the theory is presented in Section 2.3.2. Although the trajectory equations are used here to define dispersion, there are other potential applications, which include incorporation in operational Fire Control computers, the computation of firing tables, as well as any other situation requiring rapid, accurate, low-cost trajectory determination.

2.3.1 Closed Form Rationale

One approach to determining the Projectile Trajectory would be to perform full numerical simulations of the six degrees of freedom (6 DOF) differential equations of motion. However, since a Monte Carlo approach requires at least hundreds and possibly thousands of simulations in order to determine the dispersion accurately, the comparatively long running times of 6 DOF simulations makes this approach impractical. Another approach is to develop approximate trajectory equations which can be rapidly evaluated, while at the same time include the effects of the major error sources. HITS uses the latter approach. The projectile equations of motion were simplified and the trajectory was determined in closed form. Perturbation equations are solved to refine the trajectory model. Six degrees of freedom simulations were conducted to authenticate the trajectory model approximations and are presented in Appendix C. In all cases the trajectory model checked out to within engineering tolerances.

2.3.2 Summary Description

The trajectory equations are composed of three parts, as illustrated in Figure 2-2. The first part is a particle trajectory which accounts for a large number of drag effects and the effect of constant velocity winds. The second part is a perturbation model which evaluates the effect of lift on crossrange dispersion. The final part is a perturbation model to account for the downrange effect of lift induced drag. The particle trajectory is evaluated first. Corrections for lifting effects are then computed via the perturbation equations and applied to the particle trajectory. The following paragraphs discuss the three components of the Trajectory Module.

<u>Particle Trajectory</u>

The particle trajectory equations shown in Figure 2-2 account for the downrange effects of muzzle velocity, projectile weight, drag, etc., as well as crossrange and downrange winds. The equations assume a zero angle of attack, a uniform density atmosphere, and the absence of gravity, and a variable drag coefficient. Classically, particle trajectories have approximated the drag coefficient as being independent of velocity. Although adequate for conventional muzzle velocities and short flight times, the drag coefficient model for hypervelocity projectiles must include the variation with velocity in order to avoid substantial errors, since the drag coefficient of typical hypervelocity projectiles (i.e., slender bodies) can vary by a factor of 3 to 4 over the velocity range of interest. The HITS particle trajectory models the drag coefficient as

$$C_D = C_{D_\infty} + \frac{K_D}{V^2} \qquad (2.3\text{-}1)$$

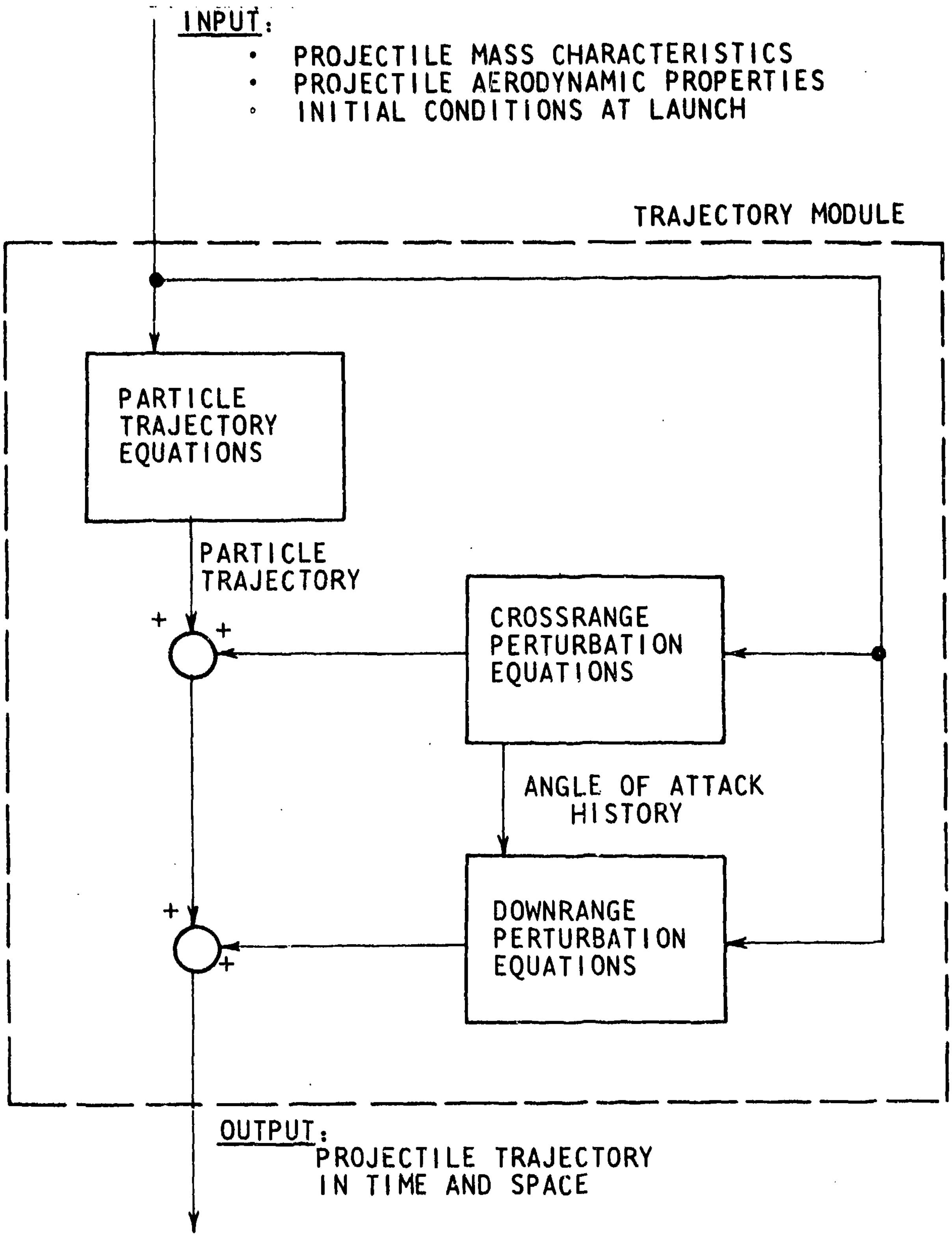


Figure 2-2 Trajectory Equations Computational Procedure

which can also be expressed in terms of the ballistic coefficient ($\beta = W/C_DA$):

$$\frac{1}{\beta} = \frac{1}{\beta_\infty} + \frac{K}{V^2} \qquad (2.3-2)$$

where $C_{D_{OO}}$ and K_D or β_{OO} and K are empirically determined constants. This is an excellent model in the subsonic and supersonic velocity regimes. Ablative effects (i.e., tip recession and mass loss) can be approximated by making appropriate changes to the drag model parameters. Thus, including the drag variation with velocity opens the door to considering the effects of ablation.

Crossrange Perturbation Equations

This component of the Trajectory Model computes perturbations to the particle trajectory due to spurious lift forces caused by angle of attack oscillations and static trim angles. The angle of attack history is determined. The effect of a constant projectile spin rate is included. Velocity is assumed to be constant, so the crossrange error arising from transient roll resonance is not determined. Aerodynamic crossrange perturbation forces are significant only during the first few pitch oscillations. Since these occur near the muzzle while velocity is essentially constant, the constant velocity assumption is realistic. The crossrange perturbations are applied to the particle trajectory as a function of range. The particle trajectory includes the variation in velocity. The solution of these equations are applied as corrections to the particle trajectory as indicated in Figure 2-2.

Downrange Perturbation Equations

The downrange perturbation equations incorporate the effects of angle of attack variations on the drag coefficient. They include the effect of a constant spin rate. During the period when the projectile is oscillating, the perturbations to the velocity and flight time are determined by computing a mean drag coefficient with and without the angle of attack history as determined by the crossrange perturbation equations. The perturbations are applied as corrections to the velocity and flight time as determined by the (zero angle of attack) particle trajectory. The trajectory is carried beyond the point of angle of attack convergence by restarting the particle trajectory solution with the corrected velocity and time at convergence as initial conditions.

2.4 Related Topics

This section discusses three subjects which are peripheral to HITS but pertinent to projectile dispersion analyses. Section 2.4.1 presents the various statistics customarily used to describe crossrange dispersion. Correlation analysis as related to error source models, is discussed in Section 2.4.2. Section 2.4.3 considers the effects of engagement geometry on target-fixed crossrange dispersion. Each of these topics has a bearing on the usefulness of the HITS code.

2.4.1 Crossrange Dispersion Indices

Crossrange dispersion is an important measure of accuracy, particularly for engagements involving slowly moving targets and [illegible] weapons. [illegible]ately, crossrange dispersion [illegible] This section defines the [illegible] and presents conversion factors. [illegible] detail.

[illegible] used to summarize cross-[illegible]

- [illegible] standard deviation is denoted by σ_{CR}.
- [illegible] is the radius of [illegible]

$$P\left[\sqrt{x^2 \ldots}\right] \ldots \qquad (2.4\text{-}1)$$

- The radius of the 80% circle is denoted R80.
- The Radial Standard Deviation (RSD) is the rms value of the impact point from the aim point, (x, y),

$$RSD = \sqrt{E\left[x^2 + y^2\right]} \qquad (2.4\text{-}2)$$

where E denotes the ensemble average.

- The "jump angle", $\bar{J}$, is the average displacement of the impact point from the aim point, (x, y),

$$\bar{J} = E\left[\sqrt{x^2 + y^2}\right] \qquad (2.4\text{-}3)$$

when the impact coordinates x and y are expressed as angular deflections.

- There is a 50% probability the impact point coordinates relative to the aim point, (x, y), will lie between two parallel lines which are equidistant from the origin and are separated by twice the Linear Error Probable (LEP).

The aim point is usually defined as the centroid of the impact points.

Since the components of crossrange dispersion can usually be assumed to have equal magnitude and be independent Gaussian random phenomenon, the statistical dispersion indices are proportional. The scale factors are presented in Figure 2-3. The HITS code evaluates the crossrange standard deviation, σ_{CR}. The probability of occurrence associated with each index is given in Table 2-1.

2.4.2 Correlation Analysis - An Example

The error source model of Table 1-2 treats each source as an independent random phenomenon. Whereas this is a perfectly valid assumption in most instances, the potentiality exists that there are some subtle correlations (i.e., interrelationships) which could affect the dispersion statistics. The HITS code can evaluate the effects of correlation provided the correlation coefficient can be determined. This section illustrates analytic procedures for evaluating correlation coefficients from design information. The most direct approach would be to use ballistic range test data to evaluate the correlation. However, for the purposes of illustration, it is assumed this avenue is not open and the correlation must be evaluated from design information. The example is incomplete in the sense that no concrete conclusions are drawn. The objective is to illustrate that correlation coefficients are amenable to analyses based on physical considerations. The correlation coefficient is theoretically discussed in Appendix A.3.3.

Intuitively, muzzle velocity variations, δV and projectile weight variations, δW_p, would be expected to be negatively correlated to some extent, since a heavier projectile should result in a lower muzzle velocity. These two sources are typically large contributors to downrange dispersion if they are considered uncorrelated. However, because a heavier projectile would be expected to slow down less, it might be suspected that when correlation is taken into account, weight and muzzle velocity effects compensate, and result in significantly less net downrange dispersion. Since they are likely to be correlated and their

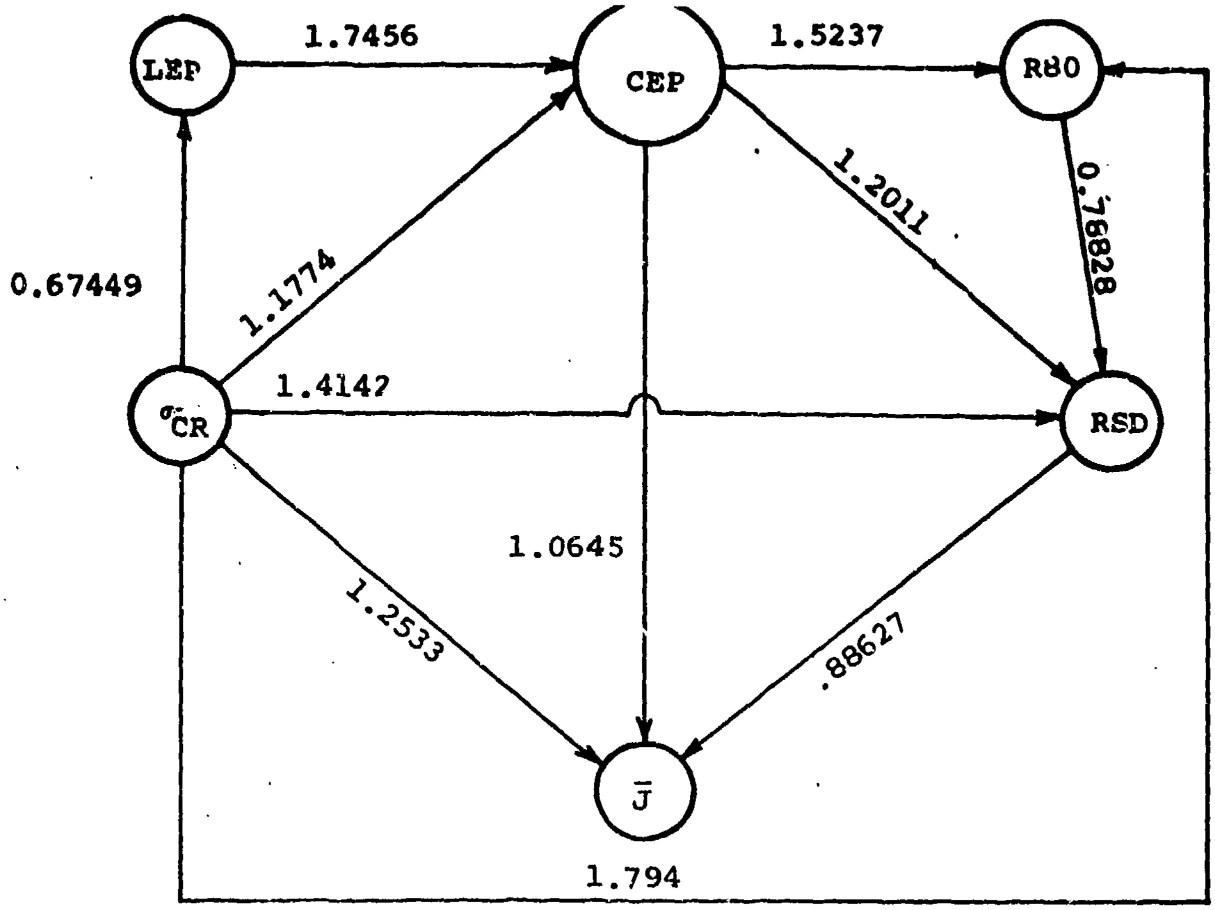


Figure 2-3 Gaussian Scale Factors

Table 2-1 Probabilities of Occurrence

Circle Radius	Probability of Occurrence (%)
LEP	20.3
σ_{CR}	39.3
CEP	50.0
$\bar{J}$	54.4
RSD	63.2
R80	80.0

correlation potentially could have a substantial effect on down-range dispersion, muzzle velocity and projectile weight variations are selected for the purpose of discussing analyses procedures.

Presuming the gun is a relatively uniform accelerator suggests an analysis of the kinetic energy imparted to the launch package might be the most direct approach to determining the correlation between projectile weight and muzzle velocity variations. (An analysis of the momentum imparted by launch might be a more fruitful approach in a launcher with an impulsive launch cycle.) The kinetic energy of the sabot/projectile is

$$K_e = \frac{1}{2} \frac{W_p + W_s}{g} V^2 \tag{2.4-4}$$

where W_s is the sabot weight and g is the acceleration due to gravity. Viewing weight as the dependent variable, Eq. (2.4-4) can be linearized about the nominal launch conditions and solved for the variation in velocity

$$\frac{\delta V}{V} = \frac{1}{2} \left\{ \frac{\delta K_e}{K_e} - \left[\eta \frac{\delta W_p}{W_p} + (1 - \eta) \frac{\delta W_s}{W_s} \right] \right\} \tag{2.4-5}$$

where η is the design ratio of projectile weight to the total package weight. Equation (2.4-5) determines the muzzle velocity-projectile weight correlation coefficient. It is

$$\rho_{VW_p} = \frac{1}{2} \left\{ \frac{\sigma_{K_e}/K_e}{\sigma_V/V} \rho_{K_e W_p} - \left[\eta \frac{\sigma_{W_p}/W_p}{\sigma_V/V} + (1-\eta) \frac{\sigma_{W_s}/W_s}{\sigma_V/V} \rho_{W_s W_p} \right] \right\} \tag{2.4-6}$$

The correlation coefficient $\rho_{W_s W_p}$ is the correlation coefficient of the projectile and sabot weights. Since these are the result of different manufacturing processes, $\rho_{W_s W_p}$ is most probably zero. σ_{W_p} and σ_V are the standard deviations of the projectile weight and the muzzle velocity and are given by the basic error source model. σ_{K_e} is the standard deviation of the kinetic energy imparted to the launch package. Assuming the launch cycle is efficient, σ_{K_e} should be dominated by the variations in the chemical energy released by the propellant. (Other factors influencing σ_{K_e} would include heat absorbed by the gun, the energy lost by in-barrel balloting and friction, and the residual energy in the gun gases.) The launch package kinetic energy-projectile weight correlation coefficient, $\rho_{K_eW_p}$, is an interesting one. At first these two quantities would appear to be unrelated ($\rho_{K_eW_p} = 0$). However, the projectile weight has an influence on the diameter of the projectile which in turn affects the tightness of the seal between the launch package and the barrel. By this argument an increased weight should cause more kinetic energy to be imparted to the launch package. Analysis of the projectile geometry and consideration of typical variations in material mass densities would determine the appropriate value of $\rho_{K_eW_p}$. All values would be substituted into Eq. (2.4-5) to determine the projectile weight-muzzle velocity correlation coefficient.

This example has illustrated the determination of correlation coefficients. They may be evaluated from ballistic range test data or design analyses. The analytical procedures are straightforward, systematic, and scientific and are based on the first principles of the physical sciences.

2.4.3 Target-Fixed Crossrange Dispersion

The HITS code defines projectile dispersion in a coordinate frame fixed with respect to the gun implacement. The dispersion must be transformed into a frame moving with the target in order to assess weapon system effectiveness. The conversion to target-fixed coordinates is usually performed during an engagement analysis and can be quite elaborate. This section presents a simplified procedure which allows the analyst to quickly determine the approximate target-fixed crossrange dispersion.

For the intercept geometry of Figure 2-4, the target-fixed crossrange dispersion in milliradians, σ_{TCD}, is given by

$$\sigma_{TCD} = \sqrt{\sigma_{CR}^2 + (S_{C/D}\,\sigma_{DR})^2} \qquad (2.4\text{-}7)$$

where σ_{CR} and σ_{DR} are the crossrange and downrange projectile dispersion standard deviations in milliradians and percent of range, respectively, and

$$S_{C/D} = \left| \frac{10 \sin\psi}{\dfrac{V_P}{V_t} + \cos\psi} \right| \qquad (2.4\text{-}8)$$

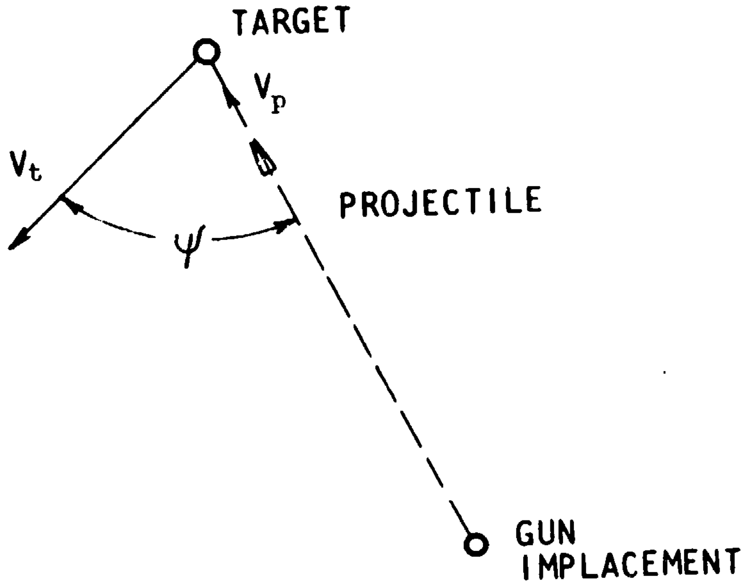


Figure 2-4 Intercept Geometry

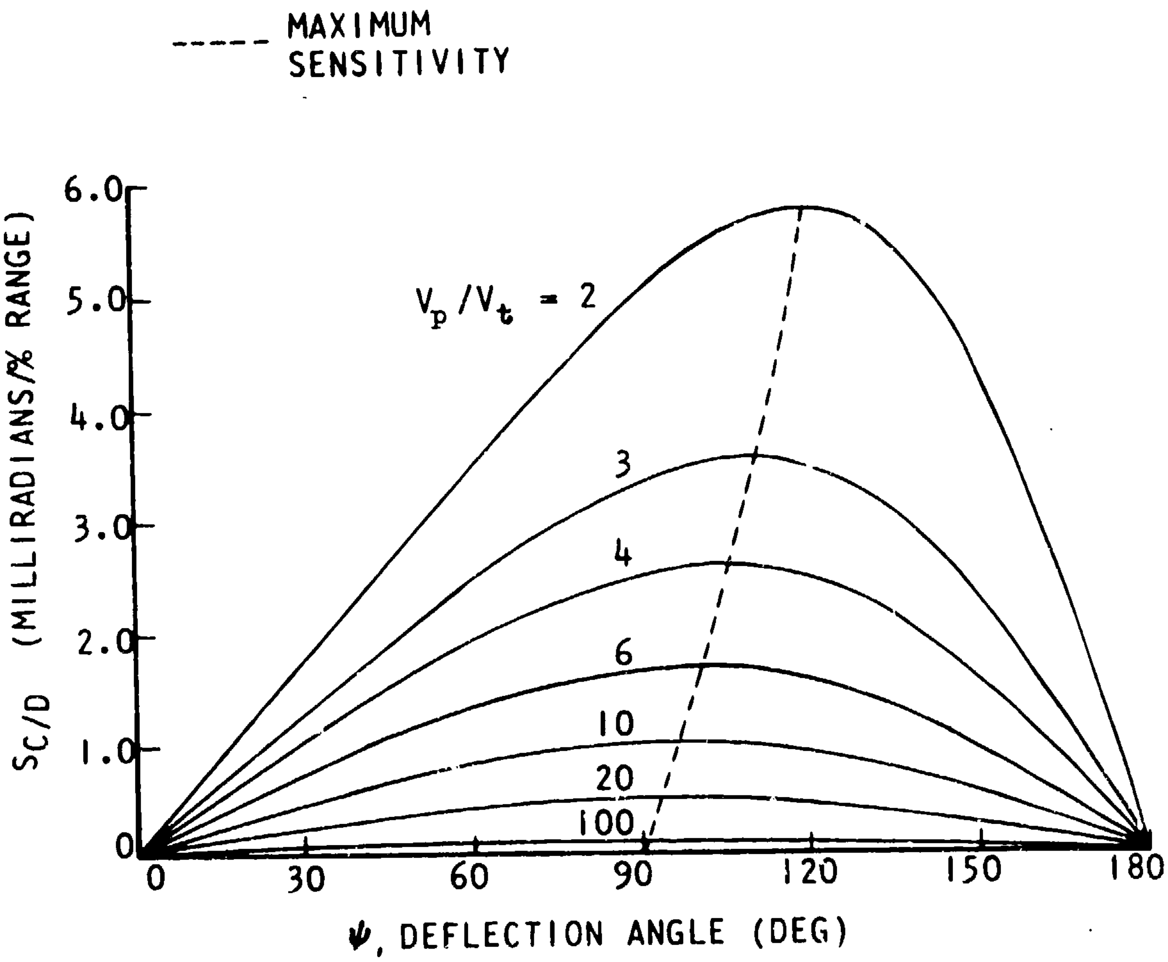


Figure 2-5 Crossrange Sensitivity to Downrange Dispersion

where V_p/V_t is the projectile to target velocity ratio in the vicinity of the target, and ψ is the deflection angle, i.e., the angle between the target velocity vector and the bore sight.

The coefficient $S_{C/D}$ is the equivalent crossrange dispersion sensitivity to downrange dispersion. The sensitivity coefficient is evaluated in Figure 2-5 for representative velocity ratios. Maximum sensitivity occurs at a deflection angle of

$$\psi_{max} = 90° + \sin^{-1}\left(\frac{V_t}{V_p}\right) \qquad (2.4\text{-}9)$$

The maximum sensitivity is

$$(S_{C/D})_{max} = \frac{10}{\sqrt{(V_p/V_t)^2 - 1}} \qquad (2.4\text{-}10)$$

and can be used to perform a worst-case analysis.

This section has presented a simple approximate technique for converting projectile dispersion into a target-fixed coordinate frame. The results may be used in conjunction with target vulnerability, projectile terminal ballistics, and target lethality data to estimate the effect of projectile dispersion on weapon system effectiveness.

3.0 INPUT DESCRIPTION

This chapter describes the order and content of the input cards. The input for a single case consists of three card groups. Section 3.1 defines the Statistical Processor Control Card (Card Group 1). Input Processor Control Cards (Card Group 2) are discussed in Section 3.2. The Optional Input Processor Control Cards (Card Group 3) are described in Section 3.3. The card groups are defined by cook-book tables that give the card format and options. Comments in the tables alert the user to related, indepth discussions presented in other portions of the text. Section 3.4 illustrates encoding of an error source model.

The card groups for a single case must appear in the order suggested by the card group numbers. The card formats utilize fixed numeric fields to facilitate data base operations. IBM-029 keypunch control cards are presented to simplify keypunching. All integer variables must be right-hand justified. Multiple cases may be stacked. Each case in the stack must be complete in itself and not simply changes from the prior case.

3.1 Statistical Processor Control Card (Card Group 1)

The Statistical Processor Control Card is the first card appearing in the deck describing a single case. This card directs the Statistical Processor to enter a specific mode of operation. Table 3-1 defines the format of the card and the various options. Figure 3-1 illustrates the associated IBM-029 keypunch control card.

The Statistical Processor Control Card (Card Group 1) described in Table 3-1, defines 13 variables. Only IØPRNT and MCØPT play significant roles and are discussed in depth. IØPRNT and MCØPT control the mode of the Statistical Processor as described in the following two paragraphs.

Referring to Table 3-1, the IØPRNT variable serves a dual purpose. When IØPRNT = 0, the Statistical Processor identifies the case as either a Single Trajectory or Range Check simulation, and relinquishes control to the Input Processor. The Input Processor determines the proper mode from the contents of Card Group 2. Whenever IØPRNT $\geq$ 0, the Statistical Processor enters the Analytical Statistical mode and requests the Input Processor to supply the contents of Card Group 2.

Table 3-1 Statistical Processor Control Card (Card Group 1)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
1	IY	Dispersion analysis selector IY = 1 procedure described in this manual.	• HITS has facilities to access (as yet undeveloped) alternate dispersion analysis procedures.
4	IØPRNT	Analytical Statistical mode calculation control. IØPRNT = 0 for Single Trajectory or Range Check modes = 1 calculation stops after limiting cases. = 2 calculation stops after unmixed first and second derivatives are evaluated. = 3 calculation stops after second order mean and first order variance are calculated. = 4 calculation stops after second order mixed partial derivatives are evaluated = 5 *calculation stops after second order mean and first order variance are calculated for correlated independent variables. = 6 calculation stops after second order variance calculation, assumes all independent variables are Gaussian. = 7 *calculation stops after second order variance for correlated independent variables, assumes all independent variables are Gaussian.	• IØPRNT ≥ 3 for Monte Carlo mode • The number of TYPE = 2, 3, and 4 variables (K234) and the number of dependent variables (KD) must be limited so that $KT\O T \leq 5000$ where: $KT\O T = KD(1+2K234)$, $I\O PRNT \leq 2$ $KT\O T = KD(3+2K234)$, $I\O PRNT \leq 3$ $KT\O T = KD\left[3+K234\left(2+\frac{K234-1}{2}\right)\right]$, $I\O PRNT \geq 4$ <u>Otherwise</u>, the C array will be exceeded and computation will terminate abnormally.

*Card Group 3 required

Table 3-1 Statistical Processor Control Card (Card Group 1) (cont'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
7	ISPRNT	Trajectory Module output suppression code ISPRNT = 0 trajectory module output suppressed = 1 trajectory module output printed at each reference to trajectory module	• ISPRNT = 1 provides extensive information. Usually used only in Single Trajectory or Range Check mode. • ISPRNT = 0 recommended for all Monte Carlo mode and most Analytical Stastical mode operations.
12	MCØPT	Monte Carlo mode control variable MCØPT = 0 no Monte Carlo experiments = 1 Monte Carlo mode to be exercised	
17	MCALC	Monte Carlo control variable to bypass Trajectory module MALC = 0 trajectory module specified by IY variable is used to evaluate all trajectories. = 1, 2, 3 a Taylor series approximation with (1) first order, (2) second order unmixed, and (3) full second order partial derivatives is used to approximate the full trajectory equations.	• MCALC = 0 recommended for all calculations. • MCALC = 1, 2, or 3 provides a more rapid evaluation at the expense of accuracy.
21-22	NCELL	Number of histogram cells for each independent and dependent variable. NCELL $\leq$ 20	• See Appendix A.6.1 for guidelines.
24-27	NTRIAL	Number of Monte Carlo experiments to be conducted	• See Appendix A.6.2 for guidelines.

Table 3-1 Statistical Processor Control Card (Card Group 1) (concl'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
29-32	IRJ1	Maximum number of Monte Carlo experiments to be rejected as outside internal ranges (predicted by Analytical Statistical mode) before computation is suspended.	• IRJ1 should be large enough to allow for 10% NCELL rejections but small enough to abort simulations with gross input errors.
34-37	IRJ2	Maximum number of Monte Carlo experiments to be rejected as outside external ranges (TYPE = 8 variable) limits before computation is suspended.	• See IRJ1 comment.
42	MCPRNT	Monte Carlo mode Trajectory Module summary print control MCPRNT = 0 suppress summary trajectory information for all Monte Carlo experiments. = 1 print summary information for each trajectory calculation.	• Summary print should be suppressed if large numbers of Monte Carlo experiments are performed.
44-47	IRRANØ	Random number generator seed for Monte Carlo mode.	• IRRANØ should be a large odd positive integer such as 1201.
52	IØPLØT	Control variable for automatic histogram bar chart plots. IØPLØT = 0 no plots = 1 plot histograms	• Plot facility is presently incomplete. Plot arrays are listed on the printed output. See Appendix B.2.5 for format.
57	ISPLØT	Automatic histogram plot format control. ISPLØT = 0 generate individual plots. = 1 overlay internal and external range histograms.	• Presently has no effect on execution.

NOTE: All Toggle Switches Up

Drum Card

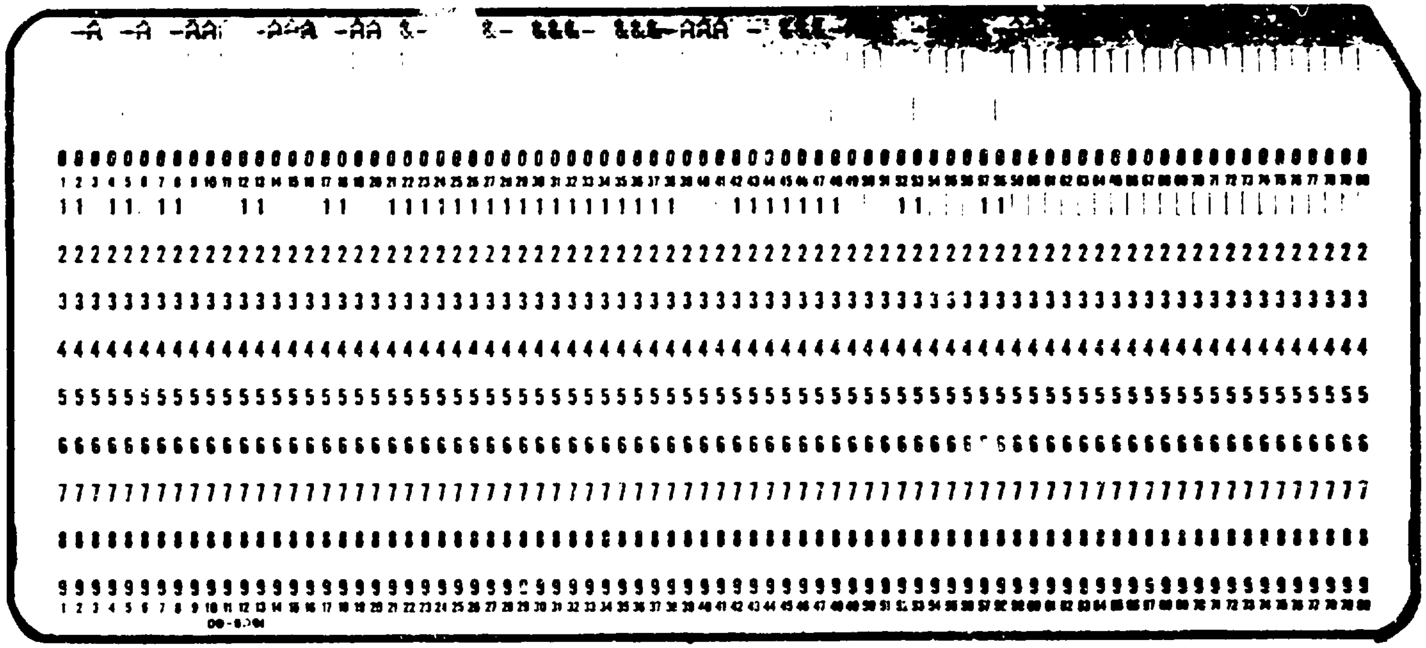


Figure 3-1 IBM-029 Keypunch Control Card for Card Group 1

The MCØPT variable of Table 3-1 determines whether or not the Statistical Processor will enter the Monte Carlo mode after completing Analytical Statistical mode calculations. If MCØPT = 1, the Statistical Processor automatically shifts into the Monte Carlo mode after the Analytical Statistical mode results have been used to set up the histograms.

3.2 Input Processor Control Cards (Card Group 2)

The Input Processor Control Cards appear second in the deck describing a single case. Each card directs the Input Processor to assign qualities to a trajectory variable. These qualities affect the calculations performed as described in Section 3.2.1. The trajectory variables are defined in Section 3.2.2.

3.2.1 Card Group 2 Assigned Qualities

Each Group 2 card defines two parameters: CØDE# and TYPE. CØDE# (i.e., code number) identifies the trajectory variable to which the qualities determined by the value of TYPE are to be assigned. This is one of the more interesting aspects of the HITS code. There are two basically different TYPE specifications that can be assigned. They are described in the succeeding paragraph. It is assumed the user wishes to determine the effect of projectile weight variations on downrange dispersion at nominal time, for the purposes of discussion.

Whenever $1 \leq$ TYPE ≤ 5, the variable is assigned the distinction of being an independent variable, that is, an independent variable whose value is determined by input data rather than preset (or default) values. In the example, projectile weight is the independent variable and would be assigned a TYPE between 1 and 5 depending on whether it is to be treated as a deterministic constant (different than the preset), a range check variable, or a random variable. Whenever $7 \leq$ TYPE ≤ 8, the variable is defined to be a dependent variable, that is, a dependent variable for which printed output is to be produced. In the example, downrange dispersion at nominal time would be declared a TYPE = 7 or TYPE = 8 variable, since the user desires to have the dispersion displayed. Thus, the CØDE# identifies the variable and TYPE determines what is to be done with it. All independent variables not mentioned in Card Group 2 are given their preset values. All dependent variables not mentioned are computed but not printed.

A Group 2 card may require more information than just the CØDE# and TYPE to completely define the variable. Table 3-2

Table 3-2 Input Processor Control Cards (Card Group 2)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
3-5	CØDE#	Independent/dependent variable identification number.	• See Table 3-3 for definitions. • Group 2 Cards may appear in any order. • Integer variable ★ Negative value (e.g. "-1") required to indicate end of Card Group 2. • CØDE# is an address in the ØE array.
7	TYPE	Statistical characterization variable for independent and dependent variables. Independent Variables: TYPE = 1 Range Check variable to be parametrically varied over the NT values on the immediately succeeding card(s).	• Every Group 2 Card must define TYPE. • Integer variable • Maximum of three TYPE = 1 variables per Card Group 2 data set. • Minimum of one and a maximum of ten dependent variables (TYPE = 7 and TYPE = 8) per Card Group 2 data set. • TYPE = 1 variables are incompatible with TYPE = 2, 3, or 4 variables. These incompatible types should never appear in the same Card Group 2 data set. ★ Additional data is required for TYPE = 1 variables. • Card format: 5 floating point fields of length 13 (beginning in column 1) per card. • Use multiple cards if required. • NT is defined later in this table.

Table 3-2 Input Processor Control Cards (Card Group 2) (cont'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION		COMMENTS
7	TYPE (cont'd)	TYPE = 2	A Gaussian random variable	• Gaussian random variables (TYPE = 2) are discussed in Appendices A.4.2 and A.4.3.
		TYPE = 3	A uniformly distributed random variable.	• Uniformly distributed random variables (TYPE = 3) are discussed in Appendix A.4.1.
		TYPE = 4	Arbitrarily distributed random variable. Immediately succeeding cards must define the probability density function at NT points.	★ Additional data required for arbitrarily distributed (TYPE = 4) random variables. • Card format: 5 floating point fields of length 13 each (beginning in colum 1) per card. • Multiple cards will be required. • Probability Density Function (PDF) represented by two paired tables. Ordinate values (i.e., PDF values) must appear first. Corresponding abscissa values must appear second with abscissa table beginning on a new card. • PDF assumed to be zero outside range of abscissa table, so use enough points. • Abscissa table must contain equally spaced values. • NT is defined later in this table.

Table 3-2 Input Processor Control Cards (Card Group 2) (cont'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
7	TYPE (cont'd)	TYPE = 5 Deterministic variable	• TYPE = 5 variables simply override preset (default) values.
		TYPE = 6 Reserved for future use	• TYPE = 6 should not be used.
		Dependent Variables:	
		TYPE = 7 Variable whose histogram range is determined by Analytical Statistical mode.	• Histogram INTERNAL RANGE and USER RANGE are identical
		TYPE = 8 Variable for which histogram is to be constructed with a range supplied by the user.	• Histogram INTERNAL RANGE determined by Analytical Statistical mode. • Histogram USER RANGE determined by user supplied values.
9-20	VALUE	Value for variable. Subject to varying definitions	• Floating point variable
		TYPE = 1, 4, 7 VALUE not required. May be left blank.	
		TYPE = 2, 3 VALUE is the expected or mean value of the random variable.	• Interpretation given in Appendix A.3 for TYPE = 2 and TYPE = 3 variables.
		TYPE = 5 VALUE is the desired deterministic value.	
		TYPE = 8 VALUE is the histogram mid-point.	
22-33	TØL	Tolerance for variable. Subject to varying definitions.	• Floating point variable.
		TYPE = 1,4,5,7 TØL not required. May be left blank.	

Table 3-2 Input Processor Control Cards (Card Group 2) (concl'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
22-33	TØL (cont'd)	TYPE = 2: TØL is the incremental length used in computing the partial derivatives.	• For TYPE = 2, TØL may be set equal to TØL = 3*STDEV
		TYPE = 3: TØL defines the variable as being uniformly distributed on the interval (VALUE-TØL, VALUE + TØL)	
		TYPE = 8: TØL defines the USERS RANGES as (VALUE-TØL, VALUE + TØL)	
35-46	STDEV	Standard deviation (σ) for some statistically defined variables.	• Floating point variable
		TYPE = 1,4,5,7,8: STDEV not required. May be left blank.	
		TYPE = 2: STDEV is the standard deviation of the Gaussian random variable.	• Interpretation given in Appendix A.3 for TYPE = 2 and TYPE = 3 variables.
		TYPE = 3: STDEV is the standard deviation of the uniformly distributed random variable.	★ For a TYPE = 3 variable uniformly distributed on the interval (VALUE-TØL, VALUE + TØL), STDEV <u>must</u> be set equal to STDEV = TØL/$\sqrt{3}$ = 0.577350*TØL
48-50	NT	Table length for additional data on immediately succeeding cards.	• Integer variable. • Succeeding cards required only for TYPE = 1 and TYPE = 4 variables.
		TYPE = 2,3,5,7,8: NT not required. May be left blank.	
		TYPE = 1: NT is the number of values over which systematic variation is to occur.	
		TYPE = 4: NT is the number of pairs of points used to represent the probability density function.	• For TYPE = 4, NT is the number of ordinate-abscissa pairs.

explains the details. In Table 3-2 hash marks are used to delimit the extent of the comments made in the right hand column. Comments identified by stars command action on the part of the user. Group 2 cards may be arranged in any order. IBM-029 keypunch control cards for Group 2 are presented in Figure 3-2.

3.2.2 Trajectory Variable Code Numbers

HITS uses "code numbers" to identify variables in input data cards and the printed output. CØDE# in Card Group 2 is just the first occurrence of code numbers in the discussion. This section discusses Table 3-3 which catalogs the code numbers.

The code number of a variable is its address in the HITS' active storage array (i.e., the ØE array). Table 3-3 lists the code numbers and their associated FORTRAN names, preset values, units, and analysis names. The code numbers also link the HITS variables to the closed form trajectory equations of Appendix C via Table C-1. An asterisk attached to a FORTRAN name denotes a potential independent variable. All others may be defined as dependent variables. The presets are the nominal values of the Table 1-2 error source model.

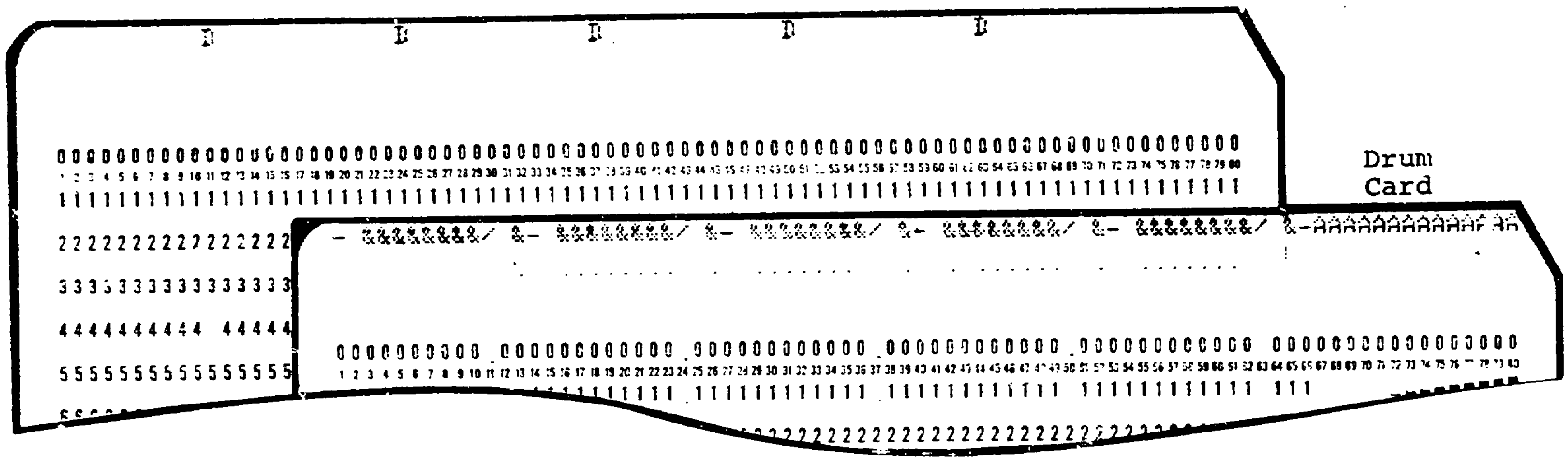
Referring to Table 3-3, the code numbers from 1 to 100 define Fire Control trajectory variables. Code numbers from 101 to 200 are the Real World trajectory variables. Variables with code numbers from 201 to 300 are system controls. These are particularly important since they can greatly reduce the amount of input data and the number of computer runs. Note that variable 203, (i.e., IFC) distinguishes between Real World and Fire Control trajectory solutions. Variables 301 to 400 are measures of dispersion. Variables 401 to 500 are computed trajectory quantities. Code numbers from 501 to 600 denote variables defining the terminal conditions of the Real World and Fire Control trajectories.

3.3 Optional Input Processor Control Cards (Card Group 3)

Optional Input Processor Control Cards (Card Group 3) appear third (last) in the deck describing a single case. Card Group 3 may not appear when the Statistical Processor is operated in either the Single Trajectory or Range Check modes. It has no effect on the Monte Carlo mode. Only the Analytical Statistical mode is affected by Group 3 data.

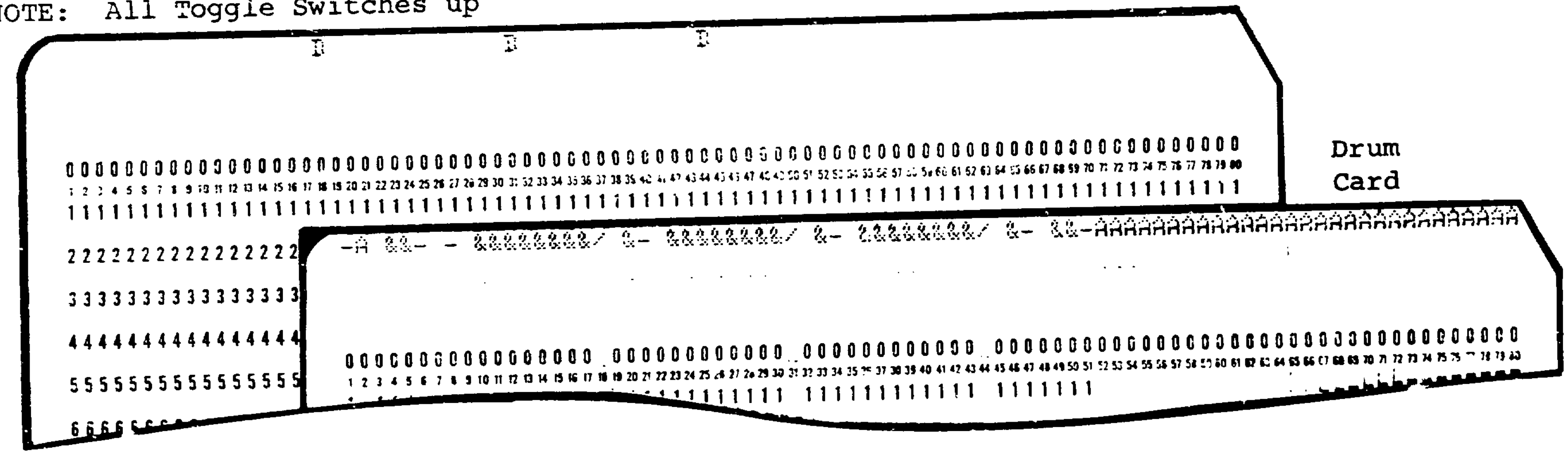
Each Group 3 card directs the Analytical Statistical Processor to treat two of the statistically defined independent

NOTE: All Toggle Switches up



(b) Additional Data Control Cards

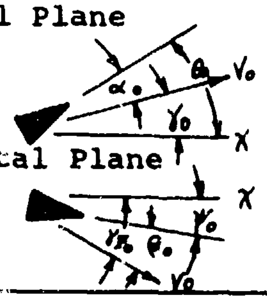
NOTE: All Toggle Switches up



(a) Basic Group 2 Control Cards

Figure 3-2 IBM-029 Keypunch Control Cards for Card Group 2

Table 3-3 HITS Trajectory Parameter List

Code Number	FORTRAN Name	Preset	Units	Remarks
1	VOF*	11,000.	ft/sec	F.C.(1) Nozzle Velocity
 2 3	 WXF* WZF*	 0. 0.	 ft/sec ft/sec	F.C. Steady Winds: Down Range Horiz. Cross Wind
4	RHØF*	2.378×10^{-3}	slugs-ft^3	F.C. Atmospheric Density
5 6 7 8	CX1F* V1F* CX2F* V2F*	3.585×10^{-2} 1.674×10^{4} 1.195×10^{-1} 3.906×10^{3}	-- ft/sec -- ft/sec	F.C. Projectile Drag Model $C_X = C_{X_\infty} + K/V^2$ Fit to (CX1F, V1F) & CX2F,V2F) CX2F < CX1F VX2F > VX1F
9	XNF*	10,000	ft	F.C. Nominal Range
 10 11 12 13 14 15	 CNAF* SMF* CMQF* CMPAF* ATRMSF* BTRMSF*	 1.9767 0.062 -7.5 0. 0. 0.	 1/rad -- -- -- deg deg	F.C. Aerodynamic Effects: • C_{N_α} • Static Margin • C_{m_q} • Magnus Moment $C_{mp\alpha}$ • Static Trim (C_m=0) -Angle of Attack -Angle of Sideslip
 16 17 18 19	 THETOF* PSIOF* TDØTOF* PDØTOF*	 0. 0. 0. 0.	 deg deg rad/sec rad/sec	F.C. Initial Attitude: θ_o Vertical Plane ψ_o $\dot{\theta}_o$ Horizontal Plane $\dot{\psi}_o$ 

*Denotes Potential Independent Variable
(1)Indicates Fire Control Parameter

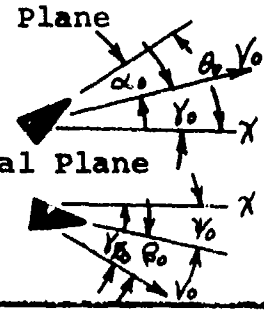
Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
				F.C. Initial Attitude (Cont'd)
20	GAMOF*	0.	deg	γ_o
21	AZOF*	0.	deg	γ_{z_o}
				F.C. Projectile Physical Characteristics:
22	AF*	3.068×10^{-3}	ft^2	A - Base Area ($\pi D^2/4$)
23	ELLF*	3.1×10^{-1}	ft	L - Length
24	DF*	6.25×10^{-2}	ft	D - Base Diameter
25	WF*	1.1×10^{-1}	lbs	W - Weight
26	AIXF*	1.235×10^{-6}	slugs-ft^2	I_x - Roll Moment of Inertia
27	AIYF*	2.110×10^{-5}	slugs-ft^2	I_y - Pitch Moment of Inertia
28	PF*	400.	rad/sec	F.C. Spin Rate
29	ALPCØF*	0.5	deg	F.C. Angle of Attack Oscillation Convergence Criterion
				F.C. Second Order Drag Effect
30	CAAF*	0.	$1/deg^2$	ΔC_x = CAAF α^2

*Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
101	V0*	11,000	ft/sec	R.W.(1) Muzzle Velocity
				R.W. Steady Winds
102	WX*	0.	ft/sec	Down Range
103	WZ*	0.	ft/sec	Horiz. Cross Wind
104	RHØ*	2.378×10^{-3}	slugs-ft^3	R.W. Atmospheric Density
				R.W. Projectile Drag Model
105	CX1*	3.585×10^{-2}	--	$C_X = C_{X_{\infty}} + K/V^2$
106	V1*	1.674×10^4	ft/sec	Fit to (CX1,V1) & (CX2,V2)
107	CX2*	1.195×10^{-1}	--	CX2 < CX1
108	V2*	3.906×10^3	ft/sec	VX2 > VX1
109	XN*	10,000	ft	R.W. Nominal Range
				R.W. Aerodynamic Effects:
110	CNA*	1.9767	1/rad	• C_{N_α}
111	SM*	0.062	--	• Static Margin
112	CMQ*	-7.5	--	• C_{m_q}
113	CMPA*	0.	--	• Magnus Moment $C_{m_{p\alpha}}$
				• Static Trim (C_m=0)
114	ATRMS*	0.	deg	-Angle of Attack
115	BTRMS*	0.	deg	-Angle of Sideslip
				R.W. Initial Attitude
116	THETO*	0.	deg	θ_o Vertical Plane
117	PSIO*	0.	deg	ψ_o
118	TDØTO*	0.	rad/sec	$\dot{\theta}_o$ Horizontal Plane
119	PDØTO*	0.	rad/sec	$\dot{\psi}_o$



*Potential Independent Variable

(1) Denotes Real World Parameter

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
				R.W. Initial Attitude (Cont'd)
120	GAMO*	0.	deg	γ_o
121	AZO*	0.	deg	γ_{z_o}
				R.W. Projectile Physical Characteristics
122	A*	$3.068\text{x}10^{-3}$	ft^2	A - Base Area ($\pi D^2/4$)
123	ELL*	$3.1\text{x}10^{-1}$	ft	L - Length
124	D*	$6.25\text{x}10^{-2}$	ft	D - Base Diameter
125	W*	$1.1\text{x}10^{-1}$	lbs	W - Weight
126	AIX*	$1.235\text{x}10^{-6}$	slugs-ft^2	I_x- Roll Moment of Inertia
127	AIY*	$2.110\text{x}10^{-5}$	slugs-ft^2	I_y-Pitch Moment of Inertia
128	P*	400.	rad/sec	R.W. Spin Rate
129	ALPCØN*	0.5	deg	R.W. Angle of Attack Oscillation Convergence Criterion
				R.W. Second Order Drag Effect
130	CAA*	0.	$1/deg^2$	$\Delta C_X = CAA\ \alpha^2$
				Externally Supplied SEP Center
131	XBART*	0.	ft	$\delta\bar{x}_T$ (Displacements from Fire Control Nominal Aim Point)
132	YBART*	0.	ft	$\delta\bar{y}_T$
133	ZBART*	0.	ft	$\delta\bar{z}_T$
				Externally Supplied CEP Center
134	YBARX*	0.	ft	$\delta\bar{y}_x$ (Displacements from Fire Control Nominal Aim Point)
135	ZBARX*	0.	ft	$\delta\bar{z}_x$

*Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
201	I[illegible]	0.	--	System Control IFCRW=1 F.C. Parameters are set equal to R.W. nominal values IFCRW=0 F.C. Parameters determined solely by inputs
202	IFCRC*	0.	--	System Control IFCRC=1 F.C. Parameters are set equal to R.W. values for each Range Check Combination IFCRC=0 F.C. Parameters determined solely by inputs.
203	IFC	1.	--	Internal Sequencing Variable IFC = 1 F.C. Solution IFC = 0 R.W. Solution
204	IDUMP*	0.	--	System Print Control IDUMP = 1 Print lots of intermediate data IDUMP = 0 Print summary data only
205	ICONV*	-10.	--	Trajectory Module Iteration Control $TOL = 10.^{ICONV}$
206	ICNCL*	1.	--	Trajectory Aerodynamic Control Parameter ICNCL = 1 C_{L_α} used in crossrange perturbation. Recommended for most applications. ICNCL = 0 C_{N_α} used in crossrange perturbation.

* Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
301	XBARTI	--	ft	Internally Computed SEP Center $\delta\bar{x}_T$ (Displacements from Fire Control Nominal Aim Point)
302	YBARTI	--	ft	$\delta\bar{y}_T$
303	ZBARTI	--	ft	$\delta\bar{z}_T$
304	YBARXI	--	ft	Internally Computed CEP Center $\delta\bar{y}_x$ (Displacements from Fire Control Nominal Aim Point)
305	ZBARXI	--	ft	$\delta\bar{z}_x$
306	DXT	--	ft	Projectile Position at Nominal Time δx_t (Displacement from Fire Control Nominal Aim Point)
307	DYT	--	ft	δy_t
308	DZT	--	ft	δz_t
309	DYX	--	ft	Projectile Position at Nominal Range δy_x (Displacement from Fire Control Nominal Aim Point)
310	DZX	--	ft	δz_x
311	DVXT	--	ft/sec	Velocity Difference at Nominal Time δV_{xt} (Difference from F.C. Value at Nominal Time)
312	DVYT	--	ft/sec	δV_{yt}
313	DVZT	--	ft/sec	δV_{zt}
314	DVXX	--	ft/sec	Velocity Difference at Nominal Range δV_{xx} Difference from F.C. Value at Nominal Time)
315	DVYX	--	ft/sec	δV_{yx}
316	DVZX	--	ft/sec	δV_{zx}
317	DT	--	sec	δt Time to Nominal Range less F.C. Nominal Time to Nominal Range

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
318	RADT[+]	--	ft	Radial Displacement from Externally Supplied SEP Center
319	RADX[+]	--	ft	Radial Displacement from Externally Supplied CEP Center
320	RHØT[(1)]	--	ft	Radial Displacement from Internally Computed SEP Center
321	RHØX[(2)]	--	ft	Radial Displacement from Internally Computed CEP Center
				Projectile Position at Nominal Time
322	DXTI	--	ft	δx_{ti} (Displacement from Internally Computed SEP Center)
323	DYTI	--	ft	δy_{ti}
324	DZTI	--	ft	δz_{ti}
				Projectile Position at Nominal Time
325	DXTE	--	ft	δx_{te} (Displacement from Externally Supplied SEP Center)
326	DYTE	--	ft	δy_{te}
327	DZTE	--	ft	δz_{te}
				Projectile Position at Nominal Range
328	DYXI	--	ft	δy_{xi} (Displacement from Internally Computed CEP Center)
329	DZXI	--	ft	δz_{xi}
				Projectile Position at Nominal Range
330	DYXE	--	ft	δy_{xe} (Displacement from Externally Supplied CEP Center)
331	DZXE	--	ft	δz_{xe}
				Correlations of Projectile Position at Nominal Time
332	DXDYT	--	ft^2	$\delta x_t\,\delta y_t$ (Displacements from Fire Control Nominal Aim Point)
333	DXDZT	--	ft^2	$\delta x_t\,\delta z_t$
334	DYDZT	--	ft^2	$\delta y_t\,\delta z_t$

[+] TYPE = 8 only

(1) Requires Code Numbers 306, 307, and 308 be TYPE = 7 or TYPE = 8.

(2) Requires Code Numbers 309 and 310 be TYPE = 7 or TYPE = 8.

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
335	DYDZX	--	ft^2	Correlation at Nominal Range $\delta y_x \delta z_x$ (Displacements from F.C. Nominal Aim Point)
				Correlations about Internally Computed SEP Center at Nominal Time
336	DXDYTI	--	ft^2	$\delta x_{ti} \delta y_{ti}$ (Displacements from Internally Computed SEP Center)
337	DXDZTI	--	ft^2	$\delta x_{ti} \delta z_{ti}$
338	DYDZTI	--	ft^2	$\delta y_{ti} \delta z_{ti}$
339	DYDZXI	--	ft^2	Correlations about Internally Computed CEP Center at Nominal Range $\delta y_{xi} \delta z_{xi}$ (Displacements from Internally Computed CEP Center)
				Correlations about Externally Supplied SEP Center at Nominal Time
340	DXDYTE	--	ft^2	$\delta x_{te} \delta y_{te}$ (Displacements from Externally Supplied SEP Center)
341	DXDZTE	--	ft^2	$\delta x_{te} \delta z_{te}$
342	DYDZTE	--	ft^2	$\delta y_{te} \delta z_{te}$
343	DYDZXE	--	ft^2	Correlations about Externally Supplied CEP Center at Nominal Range $\delta y_{xe} \delta z_{xe}$ (Displacements from Externally Supplied CEP Center)

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
400	ALPHA	--	rad	$\alpha(t_N)$ or $\alpha(x_N)$ (RW/FC)
401	ATØT	--	rad	$\bar{\alpha}(t_N)$ or $\alpha(x_N)$ (RW/FC)
402	ALPTRM	--	rad	Rolling Trim α
403	ADPHD0	--	rad/sec	$\dot{\bar{\alpha}}_o$
404	B	--	--	$\bar{\beta}$
405	BETTRM	--	rad	Rolling Trim β
406	BETA0	--	rad	β_o
407	BETAD0	--	rad/sec	$\dot{\beta}_o$
408	BETA	--	rad	$\beta(t_N)$ or $\beta(x_N)$ (RW/FC)
409	ALPMAX	--	rad	$\bar{\alpha}_{UPPER}$
410	ALPMIN	--	rad	$\bar{\alpha}_{LOWER}$
411	CAYD	--	$(ft/sec)^2$	K_D
412	CD8	--	--	$C_{D\infty}$
413	CMA	--	1/rad	C_{M_α}
414	CMTHTD	--	sec	$C_{M_{\dot{\theta}}}$
415	CMTHTA	--	sec	$C_{M_{\dot{\phi}\alpha}}$
416	CAY1	--	rad	K_1
417	CAY2	--	rad	K_2
418	CDA0B	--	--	$\bar{C}_D$ ($\bar{\alpha} = 0$)
419	CDAB	--	--	$\bar{C}_D$

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
420	DELTT	--	sec	δ_t
421	DELV	--	ft/sec	δV
422	DELT	--	sec	λ_t
423	DELX	--	ft	Δ_x
424	DYDXO	--	rad	γ_o
425	DZDXO	--	rad	γ_{z_o}
426	DYDTO	--	ft/sec	$V_y(0)$
427	DZDTO	--	ft/sec	$V_z(0)$
428	DELW	--	1/sec	$\Delta\omega$
429	DELLAM	--	1/sec	$\Delta\lambda$
430	EYEP	--	sec^2	I'
431	EMP	--	sec	m'
432	EOLT	--	--	$e^{\lambda_o t'}$
433	EDLT	--	--	$e^{\Delta\lambda_o t'}$
434	F	--	1/ft	f
435	H	--	ft/sec^2	h
436	PSIDO	--	rad/sec	$\dot{\psi}_o$
437	PHIO	--	rad	ϕ_o
438	PHI1	--	rad	ϕ_1
439	PHI2	--	rad	ϕ_2

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
440	PSIO	--	rad	ψ_o
441	R1	--	rad	R_1
442	R2	--	rad	R_2
443	R3	--	rad	R_3
444	R4	--	rad	R_4
445	RTRIM	--	--	R_{TRIM}
446	TC	--	sec	t_c
447	TCO	--	sec	t_{c_o}
448	TS	--	sec	t^*
449	TGO	--	sec	t_{G_o}
450	TP	--	sec	t'
451	THETAO	--	rad	θ_o
452	THETDO	--	rad/sec	$\dot{\theta}_o$
453	TØL	--	--	$TOL = 10.^{ICONV}$
454	VC	--	ft/sec	V_c
455	VCO	--	ft/sec	V_{c_o}
456	VYA	--	ft/sec	V_{ya}
457	VZA	--	ft/sec	V_{za}
458	WX	--	ft/sec	W_x
459	WZ	--	ft/sec	W_z

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
460	W0	--	sec^{-1}	ω_o
461	W1	--	sec^{-1}	ω_1
462	W2	--	sec^{-1}	ω_2
463	WL2	--	sec^{-2}	$2(\omega_o{}^2 + \Delta\lambda^2)$
464	XLAM0	--	sec^{-1}	λ_o
465	XLAM1	--	sec^{-1}	λ_1
466	XLAM2	--	sec^{-1}	λ_2
467	XNU1	--	rad	ν_1
468	XNU2	--	rad	ν_2
469	XG	--	ft	x_G
470	XMU	--	--	μ
471	XJAY	--	rad	JA_Y
472	XJAZ	--	rad	JA_Z
473	XJA	--	rad	JA
474	YA	--	ft	ψ_a
475	ZA	--	ft	Z_a
476	ALPH0	--	rad	a_o

Table 3-3 HITS Trajectory Parameter List (Concl'd)

Code Number	FORTRAN Name	Preset	Units	Remarks
500	TN	--	sec	F.C. Established Nominal Time at Nominal Range
				F.C. Computed Position at Nominal Time
501	XTFC	--	ft	$x_{fc}(t_N)$ (Displacements from Muzzle)
502	YTFC	--	ft	$y_{fc}(t_N)$
503	ZTFC	--	ft	$z_{fc}(t_N)$
				F.C. Computed Velocity at Nominal Time
504	VXTFC	--	ft/sec	$V_{x_{fc}}(t_N)$
505	VYTFC	--	ft/sec	$V_{y_{fc}}(t_N)$
506	VZTFC	--	ft/sec	$V_{z_{fc}}(t_N)$
				Projectile Position at Nominal Time
507	XT	--	ft	$x(t_N)$ (Displacements from Muzzle)
508	YT	--	ft	$y(t_N)$
509	ZT	--	ft	$z(t_N)$
				Projectile Velocity at Nominal Time
510	VXT	--	ft/sec	$V_x(t_N)$
511	VYT	--	ft/sec	$V_y(t_N)$
512	VZT	--	ft/sec	$V_z(t_N)$
				Projectile Position at Nominal Range
513	YR	--	ft	$y(x_N)$ (Displacements from Muzzle)
514	ZR	--	ft	$z(x_N)$
				Projectile Velocity at Nominal Range
515	VXR	--	ft	$V_x(x_N)$
516	VYR	--	ft	$V_y(x_N)$
517	VZR	--	ft	$V_z(x_N)$
518	TX	--	sec	Projectile Time to Reach Nominal Range

variables as being correlated. Thus each variable appearing in Card Group 3 must also appear in Card Group 2 with a TYPE specification of 2, 3, or 4. Table 3-4 defines the format of Group 3 cards and provides pertinent details. The comments denoted by stars command action on the part of the user. The IBM-029 keypunch control cards are presented in Figure 3-3. Group 3 cards may appear in any order.

3.4 Error Source Model Encoding

This section illustrates the encoding of the error source model of Table 1-2. Primary attention is focused on Card Group 2. Maximum instructional benefit will be achieved by the user who works the problem independently and refers to the text simply for verification.

The first task is to identify the code numbers (i.e., trajectory variable names), corresponding to line items in the error source model. Table 3-5 is a complete listing arrived at by cross referencing the error source model, Table 1-2, and the Parameter List, Table 3-3. These are Real World trajectory variables. Each of the Rayleigh/Uniform distributed sources are two-dimensional and, as a result, are described by two trajectory variables.

With the code numbers in hand, Card Group 2 may be constructed by reference to Table 3-2. For instance:

- Muzzle velocity (CØDE# = 101) is a Gaussian distributed uncertainty (TYPE = 2). The nominal or mean value is

  $$\text{VALUE} = 1.1 \times 10^{4} \text{ ft/sec}$$

  The standard deviation is

  $$\text{STDEV} = (1/3\%)(1.1 \times 10^{4}) = 3.666 \times 10^{1} \text{ ft/sec}$$

  and

  $$\text{TØL} = 3 * \text{STDEV} = 1.1 \times 10^{2} \text{ ft/sec}$$

- Projectile reference area (CØDE# = 122) is a uniformly distributed uncertainty (TYPE = 3) with a nominal or mean value of

  $$\text{VALUE} = 3.068 \times 10^{-3} \text{ ft}^2$$

Table 3-4 Optional Input Processor Control Cards (Card Group 3)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
3-5	CØDE#1	Code number for first variable of correlated pair.	• See Table 3-3 for definitions. • Group 3 cards may appear in any order. • Integer variable ★ Negative value (e.g. "-1") required to indicate end of Card Group 3. • CØDE#1 is an address in the ØE array.
8-10	CØDE#2	Code number for second variable of correlated pair. Of necessity CØDE#2 ≠ CØDE#1	• See Table 3-3 for definitions. • Integer variable • CØDE#2 is an address in the ØE array.
12-23	RHØ	The correlation coefficient between the variables defined by CØDE#1 and CØDE#2. Of necessity $-1 \le$ RHØ ≤ 1	• Floating point variable. • See Appendix A.3.3 for discussion. • The covariance matrix, Eq. (A.4-15), formed from the correlation coefficients is positive definite as theoretically required.

NOTE: All Toggle Switches up

First Card

Drum Card

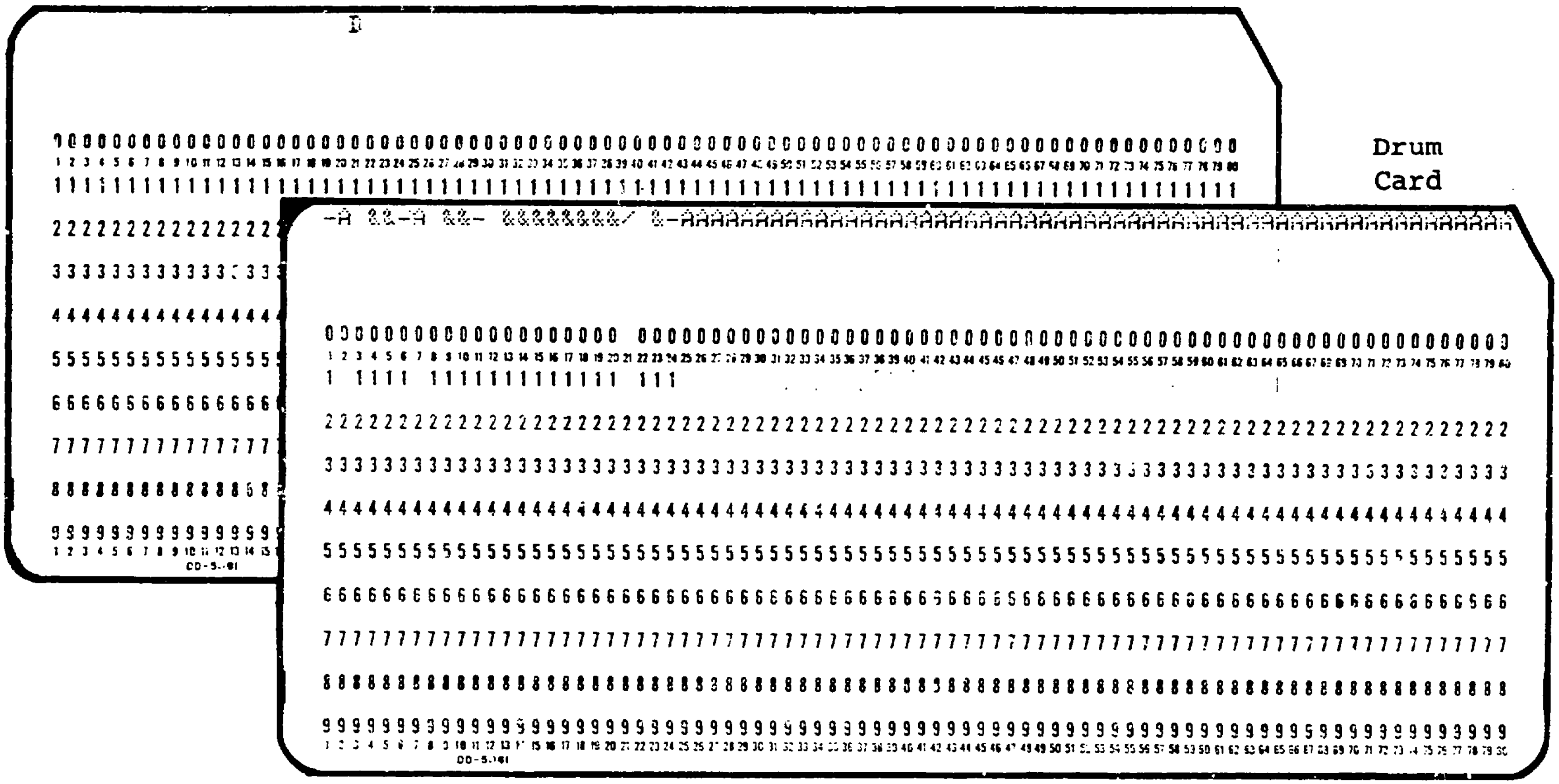


Figure 3-3 IBM-029 Keypunch Control Cards for Card Group 3

Table 3-5 Error Source Model Code Numbers

ERROR SOURCE	STATISTICAL DISTRIBUTIONS	CODE NUMBER(S)
• INITIAL CONDITIONS		
• VELOCITY VECTOR		
- MAGNITUDE	GAUSSIAN	101
- ORIENTATION	RAYLEIGH/UNIFORM*	120 & 121
• INERTIAL ORIENTATION		
- ATTITUDE	RAYLEIGH/UNIFORM*	116 & 117
- ATTITUDE RATE	RAYLEIGH/UNIFORM*	118 & 119
• PHYSICAL CHARACTERISTICS		
• WEIGHT	GAUSSIAN	125
• MOMENTS OF INERTIA		
- AXIAL	GAUSSIAN	126
- PITCH	GAUSSIAN	127
• PHYSICAL DIMENSIONS		
- REFERENCE AREA	UNIFORM	122
- LENGTH	UNIFORM	123
- BASE DIAMETER	UNIFORM	124
• AERODYNAMIC CHARACTERISTICS		
• STATIC COEFFICIENTS		
- DRAG VARIATION EFFECTS		
VELOCITY**	GAUSSIAN	105 & 107
ANGLE OF ATTACK	GAUSSIAN	130
- NORMAL FORCE	GAUSSIAN	110
• DYNAMIC COEFFICIENTS		
- PITCH DAMPING	GAUSSIAN	112
- MAGNUS MOMENT	GAUSSIAN	113
• SPIN RATE	GAUSSIAN	128
• TRIM ANGLE OF ATTACK	RAYLEIGH/UNIFORM*	114 & 115
• STATIC MARGIN	GAUSSIAN	111
• ATMOSPHERIC EFFECTS		
• CONSTANT WINDS	RAYLEIGH/UNIFORM*	102 & 103
• DENSITY VARIATIONS	GAUSSIAN	104

*Denotes a Rayleigh distribution of magnitude with a uniform 360° distribution in orientation.

**Drag variation with velocity closely approximated by $C_{X_0} = C_{X_\infty} + K_D/V^2$ which is fit to two data points $(C_{X_1}, V_1) = (0.03585, 16.740)$ and $(C_{X_2}, V_2) = (0.11951, 3906.)$. Uncertainty in C_{X_1} and C_{X_2}.

The standard deviation is

$$STDEV = (2/3\%)(3.068 \times 10^{-3}) = 2.045 \times 10^{-5} \ ft^2$$

and

$$TØL = \sqrt{3} * STDEV = 3.543 \times 10^{-5} \ ft^2$$

- Constant winds (Code Numbers 102 and 103) are Rayleigh distributed in magnitude and iniformly distributed in direction. This type of distribution is modeled as two independent Gaussian random variables (TYPE = 2). There is no average wind so

  $$VALUE = 0.0$$

  As discussed in Appendix A.4.4, the equivalent Gaussian standard deviation is given by

  $$STDEV = \frac{\bar{J}}{1.2533}$$

  where $\bar{J}$ is the mean magnitude given in the mean value column of the error source model, so

  $$STDEV = \frac{11}{1.2533} = 8.777 \ ft/sec$$

  and

  $$TØL = 3 * STDEV = 2.633 \times 10 \ ft/sec$$

These and the other error source Group 2 cards are shown in Figure 3-4.

The error source model of Table 1-2 presumed Fire Control made no simplifying assumptions in the prediction of the projectile trajectory. Thus, since Fire Control would always make use of the known nominal values, the first Group 2 card in Figure 3-4 sets a system control (CØDE# = 201) to initialize the Fire Control trajectory parameters at the nominal values of the Real World trajectory by declaring it a TYPE = 5 equal to 1. This considerably simplifies the input. Even though CØDE# = 201 denotes an integer variable, VALUE is input as a floating point number. Nominal range (CØDE# = 109) is entered as a TYPE = 5 constant value, as shown in Figure 3-4. The second order drag effect variable (CØDE# = 130) appears on two Group 2 cards. The latter prevails. The dependent variables (TYPE = 7), force

```
1   7   6   [illegible]   [illegible]   11   15)   15   15   112001   ←Card Group 1
  201  5  1.0000000D+0
  109  5  1.0000000D+4
  130  5  1.0000000D+0
  101  2  1.1000000D+4  1.1000000D+2  3.6666666D+1
  120  2  0.0000000D+0  1.1970000D-3  3.9890000D-4
  121  2  0.0000000D+0  1.1970000D-3  3.9890000D-4
  116  2  0.0000000D+0  2.3940000D-1  7.9790000D-2
  117  2  0.0000000D+0  2.3940000D-1  7.9790000D-2
  118  2  0.0000000D+0  1.3105200D+2  4.3884000D+1
  119  2  0.0000000D+0  1.3105200D+2  4.3884000D+1
  125  2  1.1000000D-1  3.3000000D-3  1.1000000D-3
  126  2  1.2340000D-6  6.1700000D-8  2.0570000D-8
  127  2  2.1100000D-5  1.0550000D-6  3.5170000D-7     Card Group 2
  122  3  3.0080000D-3  3.5430000D-5  2.0450000D-5
  123  3  3.1000000D-1  1.7900000D-3  1.0330000D-3
  124  3  6.2500000D-2  3.6080000D-4  2.0830000D-4
  105  2  3.5850000D-2  1.0760000D-3  3.5850000D-4
  107  2  1.1951000D-1  3.5850000D-3  1.1950000D-3
  13[illegible]  2  1.0000000D+0  3.0000000D-2  1.0000000D-2
  11[illegible]  2  1.9767000D+  1.1860000D-1  3.9530000D-2
  112  2  -7.5000000D+0  4.5000000D+[illegible]  1.5000000D+0
  128  2  4.0000000D+2  1.2000000D+1  4.0000000D+0
  114  2  0.0000000D+0  2.3940000D-1  7.9790000D-2
  115  2  0.0000000D+0  2.3940000D-1  7.9790000D-2
  111  2  6.2000000D-2  3.0000000D-2  1.0000000D-2
  102  2  0.0000000D+0  2.6330000D+1  8.7770000D+0
  103  2  0.0000000D+0  2.6330000D+1  8.7770000D+0
  104  2  2.3780000D-3  7.1340000D-5  2.3780000D-5
  306  7
  307  7
  308  7
   -1
  101    125  -0.2000000D+0                            } →Card Group 3
   -1
```

Figure 3-4 Encoded Error Source Model

the statistics of dispersion at nominal time to be calculated and printed. Card Group 2 is closed by the required card with a negative code number.

Card Group 2 is now complete. The basic error source model has been encoded and attention turns to selecting various HITS options. The Group 1 card is constructed by referring to Table 3-1. An Analytical Statistical mode calculation is requested by the Group 1 card in Figure 3-4. Card Group 3 calls for muzzle velocity (CØDE#1 = 101) and projectile weight (CODE#2 = 125) variations to be treated as negatively correlated. The Group 3 card was constructed in accord with Table 3-4. Card Group 3 concludes with the mandatory CØDE#1 = -1 card.

Exercising HITS with the case input of Figure 3-4 would evaluate the total dispersion at nominal time due to all twenty error sources acting in consort. If it were desired to evaluate the dispersion due to a single error source, the case input would be generated by (1) duplicating the cards shown in Figure 3-4, and (2) discarding the Group 2 cards pertaining to other error sources. Muzzle velocity and projectile weight effects would be treated as a single source, since they are declared correlated. An error budget similar to Table 1-3 could be constructed by repeating this process for each error source. The resulting case inputs could be stacked and submitted simultaneously.

4.0 OUTPUT INTERPRETATION

This chapter presents four example problems which demonstrate the basic modes of operation of the HITS code. The user is encouraged to reproduce these results to verify that the code has been properly installed and to gain familiarity.

4.1 Single Trajectory Mode

This problem demonstrates the evaluation of single-shot dispersion using the HITS code. This mode is of interest to projectile designers and exterior ballisticians.

Figure 4-1 lists the input cards. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Card Group 2 discussed in Table 3-2. Only TYPE = 5 and TYPE = 7 variables may appear in Single Trajectory mode simulations.

The results of the HITS simulation are shown in Figure 4-2. Frame 'a' is the cover sheet which identifies the code and separates the results of stacked cases. Frames 'b' through 'f' verify the inputs. Frame 'b' is the transcription of the Group 1 card. Frame 'c' lists the Card Group 2 data. Frames 'd', 'e', and 'f' are cross reference indices linking variable names and code numbers. The objective is to minimize the need to refer to Table 3-3 for variable definitions. Frame 'd' gives the FORTRAN names and code numbers of the dependent variables (TYPE = 7 and TYPE = 8). Frame 'e' gives the same information for the deterministic constants whose values have been changed from the presets (TYPE = 5 variables). All variables not changed by the input sequence are listed in Frame 'f'.

The results of calculations resented in Figure 4-2g and h. This is the information passed across the Trajectory Module interface. This level of information may be requested at any time by setting ISPRNT = 1 in Card Group 1 (see Table 3-1), although it can result in excessive output. The formats of Frames 'g' and 'h' are identical. At the top are the Fire Control parameters. These are followed by the Real World parameters, System Controls, and Computed Quantities. Table 3-3 gives the definitions of all quantities. Frame 'g' summarizes the Fire Control calculation to establish the coordinates of the nominal aim point and the nominal time-to-nominal range. The Fire Control summary is distinguished from the Real World summary by the value of the IFC System Control (IFC = 1 $\longrightarrow$ Fire Control, IFC = 0 $\longrightarrow$ Real World). Frame 'h' is the Real World trajectory. Values presented under the Computed Quantities heading quantify dispersion.

```
1   0  1
   206 5 0.0000000000
   105 5.0358539765D0
   107 5.1434159061D0
   111 5.05
   120 5-.0001
   126 5.538100000D-6
   127 5.786600000D-5
   201 51.0
   109 52085.2
   116 510.0
   318 7
   319 7
   320 7
   321 7
   306 7
   307 7
   308 7
   309 7
   310 7
    -1
```

Figure 4-1 Single Trajectory Mode Check Problem Input

```
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII   TTTTTTTTTTTTTTTTTTTT   SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII   TTTTTTTTTTTTTTTTTTTT   SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII   TTTTTTTTTTTTTTTTTTTT   SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHH            HHHH           IIII                   TTTT           SSSS
HHHHHHHHHHHHHHHHHHHH           IIII                   TTTT           SSSSSSSSSSSSSSSSSSSS
HHHHHHHHHHHHHHHHHHHH           IIII                   TTTT           SSSSSSSSSSSSSSSSSSSS
HHHHHHHHHHHHHHHHHHHH           IIII                   TTTT           SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH           IIII                   TTTT                           SSSS
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII           TTTT           SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII           TTTT           SSSSSSSSSSSSSSSSSSSS
HHHH            HHHH   IIIIIIIIIIIIIIIIIIII           TTTT           SSSSSSSSSSSSSSSSSSSS

    HYPERVELOCITY            INFLIGHT             TRAJECTORY             SCATTER


                          CCCCC  OOOOO  DDDD   EEEEE
                          C      O   O  D   D  E
                          C      O   O  D   D  EEE
                          C      O   O  D   D  E
                          CCCCC  OOOOO  DDDD   EEEEE

PREPARED BY                                        PREPARED FOR

     AVCO SYSTEMS DIVISION                            RESEARCH DIRECTORATE
     201 LOWELL STREET                                GENERAL THOMAS J. RODMAN LABORATORY
     WILMINGTON, MASSACHUSETTS 01887                  ROCK ISLAND ARSENAL
                                                      ROCK ISLAND, ILLINOIS 61201

                        EFFECTIVE DATE - JANUARY 1976
```

Figure 4-2a Single Trajectory Mode Check Problem Output

```
*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

         IY =     1     IOPRNT =     0     ISPRNT =     1

                        MONTE CARLO CONTROLS
   MCOPT =     0     MCALC =     0     NCELL =     0     NTRIAL =     0
    IRJ1 =     0      IRJ2 =     0    MCPRNT =     0     IRANNO =     0
  IOPLOT =     0    ISPLOT =     0
```

Figure 4-2b Single Trajectory Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
105	5	3.58540-02	0.0	0.0	0
107	5	1.43420-01	0.0	0.0	0
111	5	5.00000-02	0.0	0.0	0
120	5	-1.00000-04	0.0	0.0	0
126	5	5.38100-07	0.0	0.0	0
127	5	7.86600-06	0.0	0.0	0
201	5	1.00000 00	0.0	0.0	0
109	5	2.08520 03	0.0	0.0	0
116	5	1.00000 01	0.0	0.0	0
318	7	0.0	0.0	0.0	0
319	7	0.0	0.0	0.	0
320	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
305	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-2c Single Trajectory Mode Check Problem Output

```
*** DEPENDENT VARIABLES ***

RADT STATISTICAL VARIABLE    CODE NUMBER 318
RADX STATISTICAL VARIABLE    CODE NUMBER 319
RHOT STATISTICAL VARIABLE    CODE NUMBER 320
RHOX STATISTICAL VARIABLE    CODE NUMBER 321
 DXT STATISTICAL VARIABLE    CODE NUMBER 306
 DYT STATISTICAL VARIABLE    CODE NUMBER 307
 DZT STATISTICAL VARIABLE    CODE NUMBER 308
 DYX STATISTICAL VARIABLE    CODE NUMBER 309
 DZX STATISTICAL VARIABLE    CODE NUMBER 310
```

Figure 4-2d Single Trajectory Mode Check Problem Output

```
*** VARIABLES RESET BY INPUT SEQUENCE ***

  CX1 REAL WORLD          CODE NUMBER 105
  CX2 REAL WORLD          CODE NUMBER 107
   XN REAL WORLD          CODE NUMBER 109
   SM REAL WORLD          CODE NUMBER 111
THETO REAL WORLD          CODE NUMBER 116
 GAMO REAL WORLD          CODE NUMBER 120
  AIX REAL WORLD          CODE NUMBER 126
  AIY REAL WORLD          CODE NUMBER 127
IFCRW SYSTEM CONTROL      CODE NUMBER 201
ICNCL SYSTEM CONTROL      CODE NUMBER 206
```

Figure 4-2e Single Trajectory Mode Check Problem Output

```
*** VARIABLES ASSIGNED PRE[illegible] VALUES ***

     V0 FIRE CONTROL          CODE NUMBER   1
     WX FIRE CONTROL          CODE NUMBER   2
     WZ FIRE CONTROL          CODE NUMBER   3
    RHO FIRE CONTROL          CODE NUMBER   [illegible]
    CAI FIRE CONTROL          CODE NUMBER   [illegible]
     VI FIRE CONTROL          CODE NUMBER   6
    CXP FIRE CONTROL          CODE NUMBER   7
     [illegible] FIRE CONTROL          CODE NUMBER   8
     SN FIRE CONTROL          CODE NUMBER   9
    CNA FIRE CONTROL          CODE NUMBER  10
     SM FIRE CONTROL          CODE NUMBER  11
    CMQ FIRE CONTROL          CODE NUMBER  12
   CMPA FIRE CONTROL          CODE NUMBER  13
  ATRMS FIRE CONTROL          CODE NUMBER  14
  BTRMS FIRE CONTROL          CODE NUMBER  15
  THETO FIRE CONTROL          CODE NUMBER  16
   PSI0 FIRE CONTROL          CODE NUMBER  17
  TDOTO FIRE CONTROL          CODE NUMBER  18
  PDOTO FIRE CONTROL          CODE NUMBER  19
   GAMO FIRE CONTROL          CODE NUMBER  20
    AZO FIRE CONTROL          CODE NUMBER  21
      A FIRE CONTROL          CODE NUMBER  22
    ELL FIRE CONTROL          CODE NUMBER  23
      D FIRE CONTROL          CODE NUMBER  24
      W FIRE CONTROL          CODE NUMBER  25
    AIX FIRE CONTROL          CODE NUMBER  26
    AIY FIRE CONTROL          CODE NUMBER  27
      P FIRE CONTROL          CODE NUMBER  28
 ALPCON FIRE CONTROL          CODE NUMBER  29
    CAA FIRE CONTROL          CODE NUMBER  30
     V0 REAL WORLD            CODE NUMBER 101
     WX REAL WORLD            CODE NUMBER 102
     WZ REAL WORLD            CODE NUMBER 103
    RHO REAL WORLD            CODE NUMBER 104
     VI REAL WORLD            CODE NUMBER 105
     VP REAL WORLD            CODE NUMBER 108
    CNA REAL WORLD            CODE NUMBER 110
    CMQ REAL WORLD            CODE NUMBER 112
   CMPA REAL WORLD            CODE NUMBER 113
  ATRMS REAL WORLD            CODE NUMBER 114
  BTRMS REAL WORLD            CODE NUMBER 115
   PSI0 REAL WORLD            CODE NUMBER 117
  TDOTO REAL WORLD            CODE NUMBER 118
  PDOTO REAL WORLD            CODE NUMBER 119
    AZO REAL WORLD            CODE NUMBER 121
      A REAL WORLD            CODE NUMBER 122
    ELL REAL WORLD            CODE NUMBER 123
      D REAL WORLD            CODE NUMBER 124
      W REAL WORLD            CODE NUMBER 125
      P REAL WORLD            CODE NUMBER 128
 ALPCON REAL WORLD            CODE NUMBER 129
    CAA REAL WORLD            CODE NUMBER 130
  XBART REAL WORLD            CODE NUMBER 131
  YBART REAL WORLD            CODE NUMBER 132
  ZBART REAL WORLD            CODE NUMBER 133
  YBARX REAL WORLD            CODE NUMBER 134
  ZBARX REAL WORLD            CODE NUMBER 135
  IFCRC SYSTEM CONTROL        CODE NUMBER 202
  IDUMP SYSTEM CONTROL        CODE NUMBER 204
  ICONV SYSTEM CONTROL        CODE NUMBER 205
```

Figure 4-2f Single Trajectory Mode Check Problem Output

```
                              *** PROJECTILE DISPERSION CASE SUMMARY ***

                                         FIRE CONTROL PARAMETERS
      V0 =  1.10000 04 FT/SEC        WX =  0.0          FT/SEC      WZ =  0.0          FT/SEC      RHO =  2.37800-03 SL/CFT
     CY1 =  3.58540-02 (NONE)        V1 =  1.67400 04   FT/SEC     CX2 =  1.43420-01   (NONE)       V2 =  3.90600 03 FT/SEC
      XN =  2.09520 03 FT           CNA =  1.97670 00   1/RAD       SM =  5.00000-02   100/L       CMO = -7.50000 00 (NONE)
    CMPA =  0.0        (NONE)     ATRMS =  0.0          DEG      PTRMS =  0.0          DEG       THETO =  1.00000 01 DEG
    PSI0 =  0.0        DEG        TDOTC =  0.0          RAD/SC   PDOTO =  0.0          RAD/SC     GAMO = -1.00000-04 DEG
     AZ0 =  0.0        DEG            A =  3.06800-03   SQ-FT      ELL =  3.10000-01   FT            D =  6.25000-02 FT
       W =  1.10000-01 LBS          AIX =  5.38100-07   SL-SFT     AIY =  7.56600-06   SL-SFT        P =  4.00000 02 RAD/SC
  ALPCON =  5.00000-01 DEG          CAA =  0.0          1/SDEG

                                         REAL WORLD PARAMETERS
      V0 =  1.10000 04 FT/SEC        WY =  0.0          FT/SEC      WZ =  0.0          FT/SEC      RHO =  2.37800-03 SL/CFT
     CY1 =  3.58540-02 (NONE)        V1 =  1.67400 04   FT/SEC     CX2 =  1.43420-01   (NONE)       V2 =  3.90600 03 FT/SEC
      XN =  2.08520 03 FT           CNA =  1.97570 00   1/RAD       SM =  5.00000-02   100/L       CMO = -7.50000 00 (NONE)
    CMPA =  0.0        (NONE)     ATRMS =  0.0          DEG      PTRMS =  0.0          DEG       THETO =  1.00000 01 DEG
    PSI0 =  0.0        DEG        TDOTO =  0.0          RAD/SC   PDOTO =  0.0          RAD/SC     GAMO = -1.00000-04 DEG
     AZ0 =  0.0        DEG            A =  3.06800-03   SQ-FT      ELL =  3.10000-01   FT            D =  6.25000-02 FT
       W =  1.10000-01 LBS          AIX =  5.38100-07   SL-SFT     AIY =  7.56600-06   SL-SFT        P =  4.00000 02 RAD/SC
  ALPCON =  5.00000-01 DEG          CAA =  0.0          1/SDEG   XEART =  0.0          FT        YBART =  0.0        FT
   ZBART =  0.0        FT         YBARX =  0.0          FT       ZEARX =  0.0          FT

                                         SYSTEMS CONTROLS
   IFCRN =          1 (NONE)      IFCRC =           0 (NONE)       IFC =            1 (NONE)     IDUMP =            0 (NONE)
   ICONV =        -10 (NONE)      ICNCL =           0 (NONE)
                                         COMPUTED QUANTITIES
  XEARTI =  0.0        FT        YEARTI =  0.0          FT      ZBARTI =  0.0          FT       YEARXI =  0.0        FT
  ZBARXI =  0.0        FT           DXT =  0.0          FT         DYT =  0.0          FT          DZT =  0.0        FT
     DYX =  0.0        FT           DZX =  0.0          FT        DVXT =  0.0          FT/SEC     DVYT =  0.0        FT/SEC
    DVZT =  0.0        FT/SEC      DVXX =  0.0          FT/SEC    DVYX =  0.0          FT/SEC     DVZX =  0.0        FT/SEC
      DT =  0.0        SEC         RADT =  0.0          FT        RADX =  0.0          FT         RHOT =  0.0        FT
    RHOX =  0.0        FT          DXTI =  0.0          FT        DYTI =  0.0          FT         DZTI =  0.0        FT
    DXTE =  0.0        FT          DYTE =  0.0          FT        DZTE =  0.0          FT         DYXI =  0.0        FT
    DZXI =  0.0        FT          DYXE =  0.0          FT        DZXE =  0.0          FT        DXDYT =  0.0        SQ-FT
   DXDZT =  0.0        SQ-FT      DYDZT =  0.0          SQ-FT    DYDZX =  0.0          SQ-FT    DXDYTI =  0.0        SQ-FT
  DXDZTI =  0.0        SQ-FT     DYDZTI =  0.0          SQ-FT   DYDZXI =  0.0          SQ-FT    DXDYTE =  0.0        SQ-FT
  DXDZTE =  0.0        SQ-FT     DYDZTE =  0.0          SQ-FT   DYDZXE =  0.0          SQ-FT
   ALPHA =  8.42120-06 RAD         ATOT =  1.54800-05   RAD     ALPTRM =  0.0          RAD      ALPHDO = -4.04920 00 RAD/SC
       B =  1.00000 00 (NONE)    BETTRM =  0.0          RAD      BETAC =  0.0          RAD      BETADO =  6.98140 01 RAD/SC
    BETA =  1.29890-05 RAD       ALPMAX =  1.62870-05   RAD     ALPMIN = -9.98820-07   RAD        CAYD =  1.73550 06 SQ-F/S
     COA =  2.96610-02 (NONE)       CMA = -4.90220-01   1/RAD   CMTHTD = -2.13070-05   SEC      OMTHTA =  0.0        SEC
    CAY1 =  8.63770-02 RAD         CAY2 =  8.51950-02   RAD      CDAOB =  4.44700-02   (NONE)     CDAB =  4.94990-02 (NONE)
   DELTT =  1.13240-04 SEC         DELV = -3.84140 01   FT/SEC    DELT =  0.0          SEC        DELX =  0.0        FT
   DYDXO = -1.74530-06 RAD         DZDXO =  0.0          RAD     DYDTO = -1.91990-02   FT/SEC    DZDTO =  0.0        FT/SEC
    DELW =  3.96720 02 1/SEC     DELLAM = -2.[illegible]960-01 1/SEC  EYEP =  2.85140-07   SQ-SEC      EYP =  8.52040-02 SEC
    EPLT =  9.31210-05 (NONE)      EPLT =  9.80310-01   (NONE)       F =  1.06700-03   1/FT          H =  2.55200 05 FT/SC2
   PSIDO =  0.0        RAD/SEC     PHI0 = -1.60810 00   RAD       PHI1 = -1.60790 00   RAD        PHI2 = -1.60830 00 RAD
    PSI0 =  0.0        RAD           R1 =  1.71470-03   RAD         R2 =  8.63560-12   RAD          R3 = -1.71470-03 RAD
      R4 =  8.61780-02 RAD        RTRIM =  1.03460 00   (NONE)      TC =  6.22920-02   SEC         TCO =  6.21790-02 SEC
      TS =  1.37710-01 SEC          TGO =  0.0          SEC         TP =  1.89560-01   SEC      THETAO =  1.74530-01 RAD
  THETDO =  0.0        RAD/SC       TOL =  1.00000-10   (NONE)      VC =  1.06160 04   FT/SEC      VCO =  1.06540 04 FT/SEC
     VYA =  1.72470 00 FT/SEC       VZA = -6.37120-01   FT/SEC      WX =  0.0          FT/SEC       WZ =  0.0        FT/SEC
      W0 =  1.31100 03 1/SEC         W1 =  9.24700 02   1/SEC       W2 =  1.69730 03   1/SEC       WL2 =  3.43760 06 1/SQ-S
   XLAM0 = -4.99620 01 1/SEC      XLAM1 = -4.92510 01   1/SEC    XLAM2 = -4.80940 01   1/SEC      XNU1 =  1.55090 00 RAD
    XNU2 =  1.59020 00 RAD           XG =  0.0          FT         XMU =  3.80610-02   (NONE)     XJAY =  1.75840-01 RAD
    XJAZ = -6.47160-02 RAD          XJA =  1.87230-01   RAD         YA =  3.88800-01   FT           ZA = -1.34030-01 FT
   ALPHO =  1.74530-01 RAD
      TN =  2.00000-01 SEC         XTFC =  2.08520 03   FT        YTFC =  3.88800-01   FT         ZTFC = -1.34030-01 FT
   VXTFC =  9.90240 03 FT/SEC     VYTFC =  1.72470 00   FT/SEC   VZTFC = -6.37120-01   FT/SEC       XT =  0.0        FT
      YT =  0.0        FT            ZT =  0.0          FT         VXT =  0.0          FT/SEC      VYT =  0.0        FT/SEC
     VZT =  0.0        FT/SEC        YR =  0.0          FT          ZR =  0.0          FT          VXR =  0.0        FT/SEC
     VYR =  0.0        FT/SEC       VZR =  0.0          FT/SEC      TX =  0.0          SEC
```

Figure 4-2g Single Trajectory Mode Check Problem Output

COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

```
                         *** PROJECTILE DISPERSION CASE SUMMARY ***
                                   FIRE CONTROL PARAMETERS
    VO =  1.1000D 04 FT/SEC      WX =  0.0         FT/SEC       WZ =  0.0         FT/SEC       RHO =  2.3780D-03 SL/CFT
   CX1 =  3.5854D-02 (NONE)      VI =  1.5740D 04 FT/SEC       CX2 =  1.4342D-01 (NONE)         V2 =  3.9060D 03 FT/SEC
    XN =  2.0852D 03 FT         CNA =  1.9767D 00 1/RAD         SM =  5.0000D-02 100/L         CMQ = -7.5000D 00 (NONE)
  CMPA =  0.0        (NONE)   ATRMS =  0.0         DEG       BTRMS =  0.0         DEG        THETO =  1.0000D 01 DEG
  PSIO =  0.0        DEG      TDOTO =  0.0         RAD/SC    PDOTO =  0.0         RAD/SC      GAMO = -1.0000D-04 DEG
   AZO =  0.0        DEG          A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT              D =  6.2500D-02 FT
     W =  1.1000D-01 LBS        AIX =  5.3812D-07 SL-SFT       AIY =  7.5660D-06 SL-SFT          P =  4.0000D 02 RAD/SC
ALPCON =  5.0000D-01 DEG        CAA =  0.0         1/SDEG

                                   REAL WORLD PARAMETERS
    VO =  1.1000D 04 FT/SEC      WX =  0.0         FT/SEC       WZ =  0.0         FT/SEC       RHO =  2.3780D-03 SL/CFT
   CX1 =  3.5854D-02 (NONE)      VI =  1.5740D 04 FT/SEC       CX2 =  1.4342D-01 (NONE)         V2 =  3.9060D 03 FT/SEC
    XN =  2.0852D 03 FT         CNA =  1.9767D 00 1/RAD         SM =  5.0000D-02 100/L         CMQ = -7.5000D 00 (NONE)
  CMPA =  0.0        (NONE)   ATRMS =  0.0         DEG       BTRMS =  0.0         DEG        THETO =  1.0000D 01 DEG
  PSIO =  0.0        DEG      TDOTO =  0.0         RAD/SC    PDOTO =  0.0         RAD/SC      GAMO = -1.0000D-04 DEG
   AZO =  0.0        DEG          A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT              D =  6.2500D-02 FT
     W =  1.1000D-01 LBS        AIX =  5.3812D-07 SL-SFT       AIY =  7.5660D-06 SL-SFT          P =  4.0000D 02 RAD/SC
ALPCON =  5.0000D-01 DEG        CAA =  0.0         1/SDEG    XBART =  0.0         FT         YBART =  0.0        FT
 ZBART =  0.0        FT       YBARX =  0.0         FT        ZBARX =  0.0         FT

                                   SYSTEMS CONTROLS
 IFCR4 =            1 (NONE)  IFCRC =            0 (NONE)      IFC =            0 (NONE)     IDUMP =            0 (NONE)
 ICONV =          -10 (NONE)  ICNCL =            0 (NONE)
                                   COMPUTED QUANTITIES
YBARTI =  0.0        FT      YBARTI =  0.0         FT       ZBARTI =  0.0         FT        YBARXI =  0.0        FT
ZBARXI =  0.0        FT         DXT = -1.5916D-12 FT           DYT = -2.7756D-16 FT            DZT =  9.7145D-17 FT
   DYX =  0.0        FT         DZX =  0.0         FT         DVXT =  9.0949D-13 FT/SEC       DVYT = -2.2204D-16 FT/SEC
  DVZT = -4.1633D-17 FT/SEC    DVXX =  0.0         FT/SEC     DVYX =  0.0         FT/SEC      DVZX =  0.0        FT/SEC
    DT =  0.0        SEC       PADT =  1.5916D-12 FT          PADX =  0.0         FT          RHOT =  1.5916D-12 FT
  RHOX =  0.0        FT        DXTI = -1.5916D-12 FT          DYTI = -2.7756D-16 FT           DZTI =  9.7145D-17 FT
[illegible] = -1.5916D-12 FT   DYTF = -2.7756D-16 FT          DZTF =  9.7145D-17 FT           DYXI =  0.0        FT
[illegible] =  0.0   FT        DYXF =  0.0         FT         DZYF =  0.0         FT         DXDYT =  4.4176D-28 SQ-FT
 DXDZT = -1.5462D-28 SQ-FT    DYDZT = -2.6563D-32 SQ-FT      DYDZX =  0.0         SQ-FT     DXDYTI =  4.4175D-28 SQ-FT
DXDZTI = -1.5462D-28 SQ-FT   DYDZTI = -2.6563D-32 SQ-FT     DYDZXI =  0.0         SQ-FT     DXDYTF =  4.4175D-28 SQ-FT
DXDZTF = -1.5462D-28 SQ-FT   DYDZTF = -2.6563D-32 SQ-FT     DYDZXF =  0.0         SQ-FT
 ALPHA =  8.4212D-06 RAD       ATDT =  1.5480D-05 RAD       ALFTRM =  0.0         RAD       ALPHDO = -4.0497D 00 RAD/SC
     B =  1.0000D 00 (NONE)  BETTRM =  0.0         RAD        BETAO =  0.0         RAD       BETADO =  6.9814D 01 RAD/SC
  BETA =   .2989D-05 RAD     ALPMAX =  1.6287D-05 RAD       ALFMIN = -9.9882D-07 RAD         CAYO =  1.7355D 06 SQ-F/S
   CDB =  2.9551D-02 (NONE)     CMA = -4.9022D-01 1/RAD     CMTHTO = -2.1307D-05 SEC       CMTHTA =  0.0        SEC
  CAY1 =  8.6373D-02 RAD       CAY2 =  8.8195D-02 RAD        COSGE =  4.4473D-02 (NONE)      CDAR =  4.9499D-02 (NONE)
[illegible] = 1.1324D-04 SEC   DELV = -3.8614D 01 FT/SEC      DELT =  0.0         SEC        DEL_X =  0.0        FT
[illegible] = -1.7[illegible]D-04 RAD  DZDYC =  0.0    RAD   DYDTC = -1.3109D-02 FT/SEC     DZDTC =  0.0        FT/SEC
[illegible] = 7.8612D 01 1/SEC  DELLAM = -2.6949D-01 1/SEC    LYFP =  2.2514D-07 SQ-SEC   [illegible] = 2.5204D-02 SEC
[illegible] = [illegible] (NONE)  SOLT = [illegible] (NONE)      F =  1.0672D-02 1/FT     [illegible] = 2.5523D 04 FT/SC2
 PSIDD =  0.0        RAD/SC    PHI0 = -1.6081D 00 RAD         PHI1 = -1.6079D 00 RAD          PHI2 = -1.6098D 00 RAD
[illegible] = 0.0    RAD         R1 =  1.7147D-01 RAD           R2 =  8.6356D-02 RAD            R3 = -1.7147D-01 RAD
    RA =  9.8179D-02 RAD      RTRIM =  1.0946D 00 (NONE)        TC =  6.2292D-02 SEC           TC0 =  6.2179D-02 SEC
    TS =  1.3771D-01 SEC        TGC =  0.0         SEC          TP =  1.9956D-01 SEC        THETAO =  1.7453D-01 RAD
THETDO =  0.0        RAD/SC     TOL =  1.0000D-10 (NONE)        VC =  1.0616D 04 FT/SEC        VC0 =  1.0654D 04 FT/SEC
   VYA =  1.7247D 00 FT/SEC     VZA = -6.3712D-01 FT/SEC        WX =  0.0         FT/SEC        WZ =  0.0        FT/SEC
    W0 =  1.3110D 03 1/SEC       W1 =  9.2470D 02 1/SEC         W2 =  1.6273D 03 1/SEC         WL2 =  3.4376D 06 1/SQ-S
 XLAM0 = -4.9[illegible]D 01 1/SEC  XLAM1 = -4.9231D 01 1/SEC  XLAM2 = -4.9570D 01 1/SEC      XNU1 =  1.75[illegible] RAD
  XNU2 =  1.7998D 00 RAD         XG =  0.0         FT          XMU =  3.8091D-02 (NONE)       XJAY =  1.75[illegible]D-01 RAD
  YJAY = -6.431[illegible]D-02 RAD  XJA =  1.8727D-01 RAD       YA =  3.8[illegible]D-01 FT      ZA = -1.340[illegible]D-01 FT
 ALPHO =  1.7453D-01 RAD
    TN =  2.0000D-01 SEC       XTFC =  2.0852D 03 FT          YTFC =  3.8880D-01 FT          ZTFC = -1.3403D-01 FT
 VXTFC =  9.9024D 03 FT/SEC   VYTFC =  1.7247D 00 FT/SEC     VZTFC = -6.3712D-01 FT/SEC        XT =  2.0852D 03 FT
    YT =  3.8883D-01 FT          ZT = -1.3403D-01 FT           VXT =  9.9024D 03 FT/SEC        VYT =  1.7247D 00 FT/SEC
   VZT = -6.3712D-01 FT/SEC      YR =  3.8883D-01 FT            ZR = -1.3403D-01 FT            VXR =  9.9024D 03 FT/SEC
   VYR =  1.7247D 00 FT/SEC     VZR = -6.3712D-01 FT/SEC        TX =  2.0000D-01 SEC
```

Figure 4-2h Single Trajectory Mode Check Problem Output

4.2 Range Check Mode

This problem demonstrates the evaluation of dispersion for a systematic permutation of projectile characteristics. This mode could be used to define a trajectory as a function of range. The Range Check mode is of interest to projectile designers and exterior ballisticians.

Figure 4-3 illustrates the input deck for a Range Check mode simulation. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Card Group 2 discussed in Table 3-2. Only TYPE = 1, 5, and 7 variables may appear in a Range Check Card Group 2. There must be at least one TYPE = 1 variable followed by the Range Check values. There must be at least one TYPE = 7 card.

The printed output generated by HITS is illustrated in Figure 4-4. Frame 'a' is the cover sheet. Frames 'b' through 'i' document the input. Frames 'b' and 'c' were discussed in Section 4.1. Frame 'd' lists the Range Check (TYPE = 1) variables. Frames 'e' through 'h' were discussed in Section 4.1. As indicated in Frame 'h', the System Control IFCRC is left at its preset value of zero. Thus, the Fire Control parameters remain at their default values for all Range Check combinations and do not vary with each combination. If a trajectory were being generated as a function of range it would be necessary to set IFCRC = 1 (see Table 3-3 for details). Frame 'i' lists the Range Check combinations. Since there are two TYPE = 1 variables with three values each, HITS will evaluate 3 * 3 = 9 trajectories.

Frames 'j' and 'k' summarize the results of computation. The summary presents the nine combinations in a sequence of blocks with the variable appearing last in Card Group 2 varying most rapidly. The TYPE = 1 independent variables (IND. VAR.) values appear first, immediately followed by the TYPE = 7 dependent variable values. The code numbers are linked to FORTRAN names by the lists presented in Frames 'd' and 'e'.

```
1   0   0      0
  206 5  0.0000000D00
  125 1                                                                 3
 0.8000000D-1  1.1000000D-1  1.5000000D-1
  123 1                                                                 3
 2.5000000D-1  3.1000000D-1  3.5000000D-1
  105 5  0.0358000D0
  107 5  0.1434000D0
  111 5  0.0500000D0
  318 7
  319 7
  320 7
  321 7
  306 7
  307 7
  308 7
  309 7
  310 7
   -1
```

Figure 4-3 Range Check Mode Check Problem Input

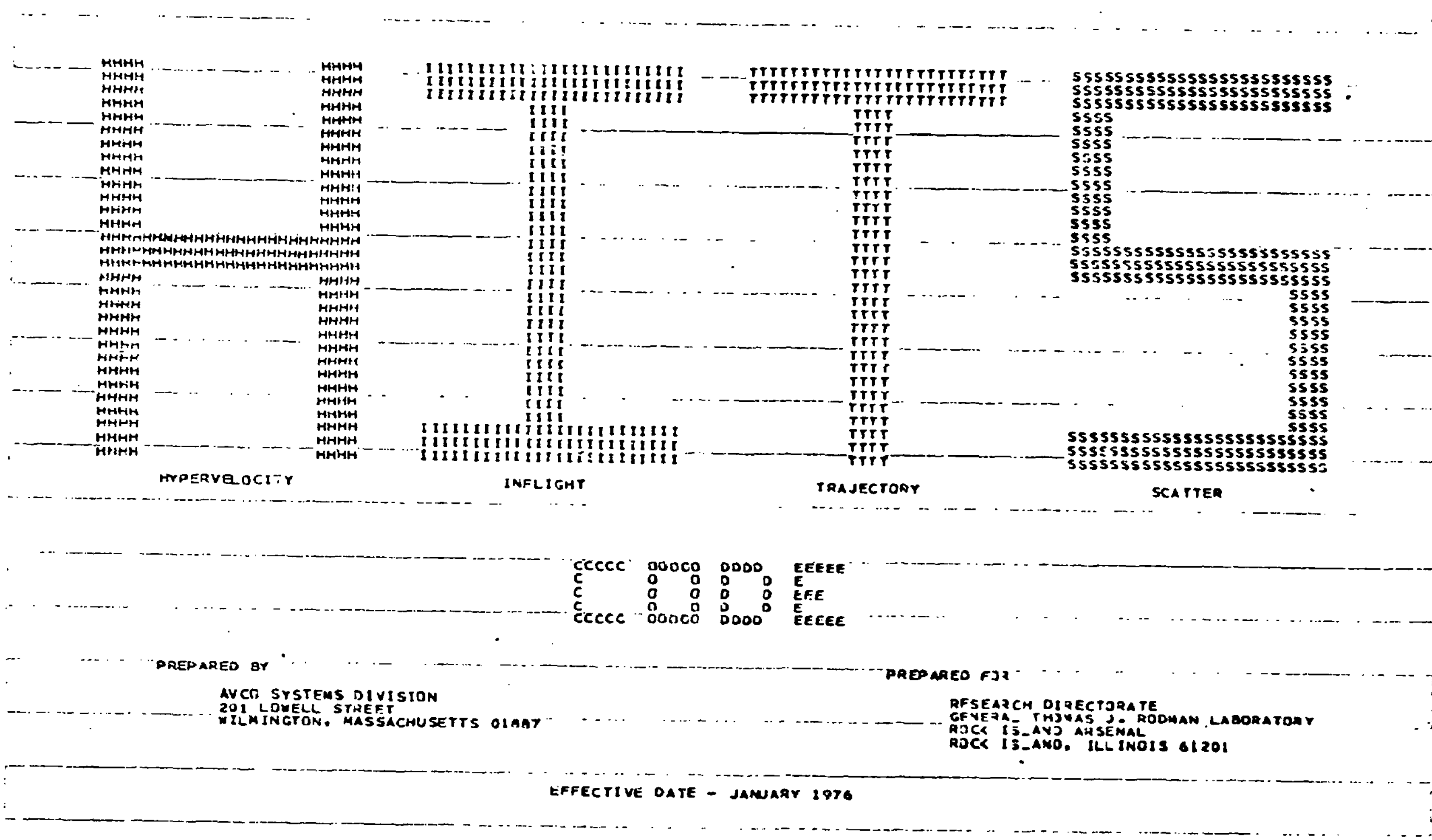


Figure 4-4a. Range Check Mode Check Problem Output

```
*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

          IY =      1     IOPHNT =     0     ISPRNT =     0

                        MONTE CARLO CONTROLS
 MCOPT =      0     MCALC =      0     NCELL =      0     NTRIAL =      0
 IRJ1 =       0     IRJ2 =       0     MCPRNT =     0     IRANNO =      0
IOPLOT =      0    ISPLOT =      0
```

Figure 4-4b Range Check Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
125	1	0.0	0.0	0.0	3
123	1	0.0	0.0	0.0	3
105	5	3.58000-02	0.0	0.0	0
107	5	1.43400-01	0.0	0.0	0
111	5	5.00000-02	0.0	0.0	0
318	7	0.0	0.0	0.0	0
319	7	0.0	0.0	0.0	0
320	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-4c Range Check Mode Check Output

```
*** RANGE-CHECK INDEPENDENT VARIABLES ***

  * REAL WORLD          CODE NUMBER 125
ELL REAL WORLD          CODE NUMBER 123
```

Figure 4-4d Range Check Mode Check Problem Output

```
*** DEPENDENT VARIABLES ***

R.. T STATISTICAL VARIABLE    CODE NUMBER 318
RADX STATISTICAL VARIABLE     CODE NUMBER 319
RHOT STATISTICAL VARIABLE     CODE NUMBER 320
RHOX STATISTICAL VARIABLE     CODE NUMBER 321
 DXT STATISTICAL VARIABLE     CODE NUMBER 306
 DYT STATISTICAL VARIABLE     CODE NUMBER 307
 DZT STATISTICAL VARIABLE     CODE NUMBER 308
 DYX STATISTICAL VARIABLE     CODE NUMBER 309
 DZX STATISTICAL VARIABLE     CODE NUMBER 310
```

Figure 4-4e Range Check Mode Check Problem Output

```
*** VARIABLES RESET BY INPUT SEQUENCE ***

 CX1 REAL WORLD          CODE NUMBER 105
 CX2 REAL WORLD          CODE NUMBER 107
  SM REAL WORLD          CODE NUMBER 111
ICNCL SYSTEM CONTROL     CODE NUMBER 206
```

Figure 4-4f Range Check Mode Check Problem Output

```
*** VARIABLES ASSIGNED PRESET VALUES ***

    WZ FIRE CONTROL          CODE NUMBER   3
   RHO FIRE CONTROL          CODE NUMBER   4
   CX1 FIRE CONTROL          CODE NUMBER   5
    V1 FIRE CONTROL          CODE NUMBER   6
   CX2 FIRE CONTROL          CODE NUMBER   7
    V2 FIRE CONTROL          CODE NUMBER   8
    XN FIRE CONTROL          CODE NUMBER   9
   CNA FIRE CONTROL          CODE NUMBER  10
    SN FIRE CONTROL          CODE NUMBER  11
   CMQ FIRE CONTROL          CODE NUMBER  12
  CMPA FIRE CONTROL          CODE NUMBER  13
 ATRMS FIRE CONTROL          CODE NUMBER  14
 BTRMS FIRE CONTROL          CODE NUMBER  15
 THETO FIRE CONTROL          CODE NUMBER  16
  PSIO FIRE CONTROL          CODE NUMBER  17
 TDOTO FIRE CONTROL          CODE NUMBER  18
 PDOTO FIRE CONTROL          CODE NUMBER  19
  GAMO FIRE CONTROL          CODE NUMBER  20
   AZO FIRE CONTROL          CODE NUMBER  21
     A FIRE CONTROL          CODE NUMBER  22
   ELL FIRE CONTROL          CODE NUMBER  23
     D FIRE CONTROL          CODE NUMBER  24
     W FIRE CONTROL          CODE NUMBER  25
   AIX FIRE CONTROL          CODE NUMBER  26
   AIY FIRE CONTROL          CODE NUMBER  27
     P FIRE CONTROL          CODE NUMBER  28
ALPCON FIRE CONTROL          CODE NUMBER  29
   CAA FIRE CONTROL          CODE NUMBER  30
    VO REAL WORLD            CODE NUMBER 101
    WX REAL WORLD            CODE NUMBER 102
    WZ REAL WORLD            CODE NUMBER 103
   RHO REAL WORLD            CODE NUMBER 104
    V1 REAL WORLD            CODE NUMBER 106
    V2 REAL WORLD            CODE NUMBER 108
    XN REAL WORLD            CODE NUMBER 109
   CNA REAL WORLD            CODE NUMBER 110
   CMQ REAL WORLD            CODE NUMBER 112
  CMPA REAL WORLD            CODE NUMBER 113
 ATRMS REAL WORLD            CODE NUMBER 114
 BTRMS REAL WORLD            CODE NUMBER 115
 THETO REAL WORLD            CODE NUMBER 116
  PSIO REAL WORLD            CODE NUMBER 117
 TDOTO REAL WORLD            CODE NUMBER 118
 PDOTO REAL WORLD            CODE NUMBER 119
  GAMO REAL WORLD            CODE NUMBER 120
   AZO REAL WORLD            CODE NUMBER 121
     A REAL WORLD            CODE NUMBER 122
    WX FIRE CONTROL          CODE NUMBER   2
     D REAL WORLD            CODE NUMBER 124
    VO FIRE CONTROL          CODE NUMBER   1
   AIX REAL WORLD            CODE NUMBER 126
   AIY REAL WORLD            CODE NUMBER 127
     P REAL WORLD            CODE NUMBER 128
ALPCON REAL WORLD            CODE NUMBER 129
   CAA REAL WORLD            CODE NUMBER 130
 XBART REAL WORLD            CODE NUMBER 131
 YBART REAL WORLD            CODE NUMBER 132
 ZBART REAL WORLD            CODE NUMBER 133
 YBARX REAL WORLD            CODE NUMBER 134
 ZBARX REAL WORLD            CODE NUMBER 135
```

Figure 4-4g Range Check Mode Check Problem Output

```
IFCRW SYSTEM CONTROL          CODE NUMBER 201
IFCRC SYSTEM CONTROL          CODE NUMBER 202
IDUMP SYSTEM CONTROL          CODE NUMBER 204
ICONV SYSTEM CONTROL          CODE NUMBER 205
```

Figure 4-4h Range Check Mode Check Problem Output

*** RANGE CHECK VALUES ***

CODE NUMBER	1 VALUE	2 VALUE	3 VALUE	4 VALUE	5 VALUE
125	8.00000-02	1.10000-01	1.50000-01		
123	2.50000-01	3.10000-01	3.50000-01		

Figure 4-4i Range Check Mode Check Problem Output

```
*** RANGE CHECK INDEPENDENT VARIABLE COMBINATIONS ***

IND. VAR.  OE( 125)=  8.0000D-02  OE( 123)=  2.5000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.0688D 03  0.0         1.0688D 03  0.0        -1.0688D 03  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  8.0000D-02  OE( 123)=  3.1000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.0688D 03  0.0         1.0688D 03  0.0        -1.0688D 03  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  8.0000D-02  OE( 123)=  3.5000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.0688D 03  0.0         1.0688D 03  0.0        -1.0688D 03  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  1.1000D-01  OE( 123)=  2.5000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.6225D 02  0.0         1.6225D 02  0.0        -1.6225D 02  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  1.1000D-01  OE( 123)=  3.1000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.6225D 02  0.0         1.6225D 02  0.0        -1.6225D 02  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  1.1000D-01  OE( 123)=  3.5000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
    1.6225D 02  0.0         1.6225D 02  0.0        -1.6225D 02  0.0         0.0         0.0         0.0

IND. VAR.  OE( 125)=  1.5000D-01  OE( 123)=  2.5000D-01

    OE( 318)    OE( 319)    OE( 320)    OE( 321)    OE( 306)    OE( 307)    OE( 308)    OE( 309)    OE( 310)
```

Figure 4-4j Range Check Mode Check Problem Output

```
5.5606D 02  0.0      5.5606D 02  0.0      5.5606D 02  0.0      0.0       0.0       0.0

IND. VAR.  DE( 125)=  1.5000D-01  DE( 123)=  3.1000D-01
  DE( 318)    DE( 319)    DE( 320)    DE( 321)    DE( 306)    DE( 307)    DE( 308)    DE( 109)    DE( 310)
  5.5606D 02  0.0         5.5606D 02  0.0         5.5606D 02  0.0         0.0         0.0         0.0

IND. VAR.  DE( 125)=  1.5000D-01  DE( 123)=  3.?0000D-01
  DE( 318)    DE( 319)    DE( 320)    DE( 321)    DE( 306)    DE( 307)    DE( 308)    DE( 309)    DE( 310)
  5.5606D 02  0.0         5.5606D 02  0.0         5.5606D 02  0.0         0.0         0.0         0.0
```

Figure 4-4k Range Check Mode Check Problem Output

4.3 Analytical Statistical Mode

This problem demonstrates the analytical evaluation of dispersion statistics. This mode is of interest to projectile designers and those concerned with the implications of projectile dispersion to overall weapon system effectiveness.

Figure 4-5 illustrates the input deck for an Analytical Statistical mode simulation. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Group 2 discussed in Table 3-2. Only TYPE = 2, 3, 4, 5, and 7 cards may appear in an Analytical Statistical mode simulation. There must be at least one TYPE = 2, 3, or 4 and one TYPE = 7 variable. Figure 4-5 illustrates the TYPE = 4 variable input format The additional data defining the probability density function must immediately follow the TYPE = 4 Group 2 card.

Figure 4-6 illustrates the Analytical Statistical mode printed output. Frames 'a' through 'h' have been described previously. Frames 'i' and 'j' list the TYPE = 4 probability density functions. Frames 'k' and 'l' give the Fire Control (IFC = 1) and Real World (IFC = 0) solutions for the nominal conditions. Frame 'm' is a self-explanatory warning which appears only when code numbers 320 and 321 are declared TYPE = 7.

Frames 'n' and 'o' of Figure 4-6 present the nominal case independent and dependent variable values, respectively. Whenever the warning doesn't appear, Frame 'o' is followed by the LIMITING CASES output illustrated in Figure 4-6, Frame 'oo', presented at the conclusion of Figure 4-6. As shown in Frame 'oo', each independent variable is incrementally varied by the TØLERANCE amount about the nominal given by VALUE. TØLERANCE is added to get the first set of ependent variable values and subtracted to get the second. Return to the example problem at Frame 'p'. These are the values of the partial derivatives of the dependent variables with respect to the independent. Independent variable code numbers are listed on the left. Dependent variable code numbers are listed above. 1ST and 2ND indicate the first and second unmixed partial derivatives. The second order mixed partial derivatives are printed next as illustrated in Frame 'q'. These are the second order partial derivatives of the dependent variables listed above with respect to the independent variables listed to the left. The partial derivatives define the sensitivity coefficients customarily listed in an error budget.

Frame 'r' of Figure 4-6 presents the dispersion statistics. This is the culmination of the Analytical Statistical mode. The heading summarizes the level of the calculation as determined by the IØPRNT variable in Card Group 1. The mean values, variances, and standard deviations of the dependent variables are printed. Standard deviations are printed a second time with greater numerical precision.

Had any of the independent variables been declared correlated by Card Group 3, Frame 'r' would have been printed assuming no correlation. The following two frames would document the effect of the correlation specified by Card Group 3. The dispersion would be presented in the format of Frame 'r'. Printing the dispersion statistics with and without correlation make it possible to determine the size of the effect from a single simulation.

```
1   4   0
  206 5  0.0000000000
  116 2  0.0000000D0   3.0000000D0   1.0000000D0
  117 2  0.0000000D0   3.0000000D0   1.0000000D0
  125 4                                                 13
0.004430OD0   0.017520OD0   0.053980OD0   0.129500OD0   0.241900OD0
0.3520300D00  0.3989000D00  0.3520300D00  0.2419000D00  0.1295000D00
0.053980OD0   0.017520OD0   0.004430OD0
 15.000000D-2  15.500000D-2  16.000000D-2  16.500000D-2  17.000000D-2
 17.500000D-2  18.000000D-2  18.500000D-2  19.000000D-2  19.500000D-2
 20.000000D-2  20.500000D-2  21.000000D-2
  123 4                                                 13
0.004430OD0   0.017520OD0   0.053980OD0   0.129500OD0   0.241900OD0
0.3520300D00  0.3989000D00  0.3520300D00  0.2419000D00  0.1295000D00
0.053980OD0   0.017520OD0   0.004430OD0
 33.000000D-2  33.500000D-2  34.000000D-2  34.500000D-2  35.000000D-2
 35.500000D-2  36.000000D-2  36.500000D-2  37.000000D-2  37.500000D-2
 38.000000D-2  38.500000D-2  39.000000D-2
  105 5  0.035800OD0
  107 5  0.143400OD0
  111 5  0.050000OD0
  320 7
  321 7
  306 7
  307 7
  308 7
  309 7
  310 7
   -1
```

Figure 4-5 Analytical Statistical Mode Check Problem Input

```
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII     TTTTTTTTTTTTTTTTTTTTTTTTT     SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII     TTTTTTTTTTTTTTTTTTTTTTTTT     SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII     TTTTTTTTTTTTTTTTTTTTTTTTT     SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHH                 HHHH               IIII                          TTTT                SSSS
HHHHHHHHHHHHHHHHHHHHHHHHH               IIII                          TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS
HHHHHHHHHHHHHHHHHHHHHHHHH               IIII                          TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS
HHHHHHHHHHHHHHHHHHHHHHHHH               IIII                          TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH               IIII                          TTTT                                     SSSS
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII               TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII               TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS
HHHH                 HHHH     IIIIIIIIIIIIIIIIIIIIIIIII               TTTT                SSSSSSSSSSSSSSSSSSSSSSSSS

      HYPERVELOCITY                  INFLIGHT                     TRAJECTORY                    SCATTER


                                        CCCCC  OOOOO  DDDD   EEEEE
                                        C      O   O  D   D  E
                                        C      O   O  D   D  EEE
                                        C      O   O  D   D  E
                                        CCCCC  OOOOO  DDDD   EEEEE


      PREPARED BY                                                 PREPARED FOR

           AVCO SYSTEMS DIVISION                                       RESEARCH DIRECTORATE
           201 LOWELL STREET                                           GENERAL THOMAS J. RODMAN LABORATORY
           WILMINGTON, MASSACHUSETTS 01887                             ROCK ISLAND ARSENAL
                                                                       ROCK ISLAND, ILLINOIS 61201


                                    EFFECTIVE DATE - JANUARY 1976
```

Figure 4-6a Analytical Statistical Mode Check Problem Output

```
*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

          IY =     1     IOPRNT =     4     ISPRNT =     0

                     MONTE CARLO CONTROLS
MCOPT =    0     MCALC =    0     MCELL =    0     NTRIAL =    0
 IRJ1 =    0      IRJ2 =    0    MCPRNT =    0     IRANNO =    0
IOPLOT =   0    ISPLOT =    0
```

Figure 4-6b Analytical Statistical Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
116	2	0.0	3.00000D 00	1.00000D 00	0
117	2	0.0	3.00000D 00	1.00000D 00	0
125	4	0.0	0.0	0.0	13
123	4	0.0	0.0	0.0	13
105	5	3.58000D-02	0.0	0.0	0
107	5	1.43400D-01	0.0	0.0	0
111	5	5.00000D-02	0.0	0.0	0
320	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-6c Analytical Statistical Mode Check Problem Output

*** STOCHASTIC INDEPENDENT VARIABLES ***

THETO REAL WORLD	CODE NUMBER 116
PSIO REAL WORLD	CODE NUMBER 117
ELL REAL WORLD	CODE NUMBER 123
W REAL WORLD	CODE NUMBER 125

Figure 4-6d Analytical Statistical Mode Check Problem Output

```
*** DEPENDENT VARIABLES ***

RHOT STATISTICAL VARIABLE     CODE NUMBER 320
RHOX STATISTICAL VARIABLE     CODE NUMBER 321
 DXT STATISTICAL VARIABLE     CODE NUMBER 306
 DYT STATISTICAL VARIABLE     CODE NUMBER 307
 DZT STATISTICAL VARIABLE     CODE NUMBER 308
 DYX STATISTICAL VARIABLE     CODE NUMBER 309
 DZX STATISTICAL VARIABLE     CODE NUMBER 310
```

Figure 4-6e Analytical Statistical Mode Check Problem Output

```
*** VARIABLES RESET BY INPUT SEQUENCE ***

        CX1 REAL WORLD              CODE NUMBER 105
        CX2 REAL WORLD              CODE NUMBER 107
         SM REAL WORLD              CODE NUMBER 111
      ICNCL SYSTEM CONTROL          CODE NUMBER 204
```

Figure 4-6f Analytical Statistical Mode Check Problem Output

```
*** VARIABLES ASSIGNED PRESET VALUES ***

    CX1 FIRE CONTROL          CODE NUMBER   5
     V1 FIRE CONTROL          CODE NUMBER   6
    CX2 FIRE CONTROL          CODE NUMBER   7
     V2 FIRE CONTROL          CODE NUMBER   8
     XN FIRE CONTROL          CODE NUMBER   9
    CNA FIRE CONTROL          CODE NUMBER  10
     SM FIRE CONTROL          CODE NUMBER  11
    CMQ FIRE CONTROL          CODE NUMBER  12
   CMPA FIRE CONTROL          CODE NUMBER  13
  ATRMS FIRE CONTROL          CODE NUMBER  14
  BTRMS FIRE CONTROL          CODE NUMBER  15
  THETO FIRE CONTROL          CODE NUMBER  16
   PSIO FIRE CONTROL          CODE NUMBER  17
  TDOTO FIRE CONTROL          CODE NUMBER  18
  PDOTO FIRE CONTROL          CODE NUMBER  19
   GAMO FIRE CONTROL          CODE NUMBER  20
    AZO FIRE CONTROL          CODE NUMBER  21
      A FIRE CONTROL          CODE NUMBER  22
    ELL FIRE CONTROL          CODE NUMBER  23
      D FIRE CONTROL          CODE NUMBER  24
      W FIRE CONTROL          CODE NUMBER  25
    AIX FIRE CONTROL          CODE NUMBER  26
    AIY FIRE CONTROL          CODE NUMBER  27
      P FIRE CONTROL          CODE NUMBER  28
 ALPCON FIRE CONTROL          CODE NUMBER  29
    CAA FIRE CONTROL          CODE NUMBER  30
     VO REAL WORLD            CODE NUMBER 101
     WX REAL WORLD            CODE NUMBER 102
     WZ REAL WORLD            CODE NUMBER 103
    RHO REAL WORLD            CODE NUMBER 104
     V1 REAL WORLD            CODE NUMBER 106
     V2 REAL WORLD            CODE NUMBER 108
     XN REAL WORLD            CODE NUMBER 109
    CNA REAL WORLD            CODE NUMBER 110
    CMQ REAL WORLD            CODE NUMBER 112
   CMPA REAL WORLD            CODE NUMBER 113
  ATRMS REAL WORLD            CODE NUMBER 114
  BTRMS REAL WORLD            CODE NUMBER 115
     VO FIRE CONTROL          CODE NUMBER   1
     WX FIRE CONTROL          CODE NUMBER   2
  TDOTO REAL WORLD            CODE NUMBER 118
  PDOTO REAL WORLD            CODE NUMBER 119
   GAMO REAL WORLD            CODE NUMBER 120
    AZO REAL WORLD            CODE NUMBER 121
      A REAL WORLD            CODE NUMBER 122
     WZ FIRE CONTROL          CODE NUMBER   3
      D REAL WORLD            CODE NUMBER 124
    RHO FIRE CONTROL          CODE NUMBER   4
    AIX REAL WORLD            CODE NUMBER 126
    AIY REAL WORLD            CODE NUMBER 127
      P REAL WORLD            CODE NUMBER 128
 ALPCON REAL WORLD            CODE NUMBER 129
    CAA REAL WORLD            CODE NUMBER 130
  X3ART REAL WORLD            CODE NUMBER 131
  Y3ART REAL WORLD            CODE NUMBER 132
  Z3ART REAL WORLD            CODE NUMBER 133
  Y3ARX REAL WORLD            CODE NUMBER 134
  Z3ARX REAL WORLD            CODE NUMBER 135
  IFCRW SYSTEM CONTROL        CODE NUMBER 201
  IFCRC SYSTEM CONTROL        CODE NUMBER 202
```


Figure 4-6g Analytical Statistical Mode Check Problem Output

```
IDUMP SYSTEM CONTROL     CODE NUMBER 204
ICONV SYSTEM CONTROL     CODE NUMBER 205
```

Figure 4-6h Analytical Statistical Mode Check Problem Output

*** CODE NUMBER 123 PROBABILITY DENSITY FUNCTION ***

POINT NUMBER	COORDINATE VALUE	PROBABILITY DENSITY
1	3.30000-01	4.43000-03
2	3.35000-01	1.75200-02
3	3.40000-01	5.39800-02
4	3.45000-01	1.29500-01
5	3.500[illegible]	2.41900-01
6	3.55000-0[illegible]	3.52030-01
7	3.60000-01	3.98900-01
8	3.65000-01	3.52030-01
9	3.70000-01	2.41900-01
10	3.75000-01	1.29500-01
11	3.80000-01	5.39800-02
12	3.85000-01	1.75200-02
13	3.90000-01	4.43000-03

COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

Figure 4-6i Analytical Statistical Mode Check Problem Output

*** CODE NUMBER 125 PROBABILITY DENSITY FUNCTION ***

POINT NUMBER	COORDINATE VALUE	PROBABILITY DENSITY
1	1.50000D-01	4.43000D-03
2	1.55000D-01	1.75200D-02
3	1.60000D-01	5.39800D-02
4	1.65000D-01	1.29500D-01
5	1.70000D-01	2.41900D-01
6	1.75000D-01	3.52030D-01
7	1.80000D-01	3.95900D-01
8	1.85000D-01	3.52030D-01
9	1.90000D-01	2.41900D-01
10	1.95000D-01	1.29500D-01
11	2.00000D-01	5.39800D-02
12	2.05000D-01	1.75200D-02
13	2.10000D-01	4.43000D-03

Figure 4-6j Analytical Statistical Mode Check Problem Output

```
                                   *** PROJECTILE DISPERSION CASE SUMMARY ***

                                           FIRE CONTROL PARAMETERS
     V0 =  1.10000 04 FT/SEC        WX =  0.0          FT/SEC      WZ =  0.0          FT/SEC       RHO =  2.37800-03 SL/CFT
    CX1 =  3.58540-02 (NONE)        V1 =  1.67400 04 FT/SEC       CX2 =  1.19510-01 (NONE)         V2 =  3.90600 03 FT/SEC
     XN =  1.00000 04 FT           CNA =  1.97670 00 1/RAD         SM =  6.20000-02 100/L         CMO = -7.50000 00 (NONE)
   CMPA =  0.0        (NONE)     ATRMS =  0.0          DEG      BTRMS =  0.0          DEG        THETO =  0.0          DEG
   PSI0 =  0.0        DEG        TDOTO =  0.0          RAD/SC   PDOTO =  0.0          RAD/SC      GAMO =  0.0          DEG
    AZ0 =  0.0        DEG            A =  3.06800-03 SQ-FT        ELL =  3.10000-01 FT              D =  6.25000-02 FT
      W =  1.10000-01 LBS          AIX =  1.23460-06 SL-SFT       AIY =  2.11030-05 SL-SFT          P =  4.00000 02 RAD/SC
 ALPCON =  5.00000-01 DEG          CAA =  0.0          1/SDEG

                                           REAL WORLD PARAMETERS
     V0 =  1.10000 04 FT/SEC        WX =  0.0          FT/SEC      WZ =  0.0          FT/SEC       RHO =  2.37800-03 SL/CFT
    CX1 =  3.58000-02 (NONE)        V1 =  1.67400 04 FT/SEC       CX2 =  1.43400-01 (NONE)         V2 =  3.90600 03 FT/SEC
     XN =  1.00000 04 FT           CNA =  1.97670 00 1/RAD         SM =  5.00000-02 100/L         CMO = -7.50000 00 (NONE)
   CMPA =  0.0        (NONE)     ATRMS =  0.0          DEG      BTRMS =  0.0          DEG        THETO =  0.0          DEG
   PSI0 =  0.0        DEG        TDOTO =  0.0          RAD/SC   PDOTO =  0.0          RAD/SC      GAMO =  0.0          DEG
    AZ0 =  0.0        DEG            A =  3.06800-03 SQ-FT        ELL =  3.60000-01 FT              D =  6.25000-02 FT
      W =  1.80000-01 LBS          AIX =  1.23460-06 SL-SFT       AIY =  2.11030-05 SL-SFT          P =  4.00000 02 RAD/SC
 ALPCON =  5.00000-01 DEG          CAA =  0.0          1/SDEG   XBART =  0.0          FT         YBART =  0.0          FT
  ZBART =  0.0        FT         YBARX =  0.0          FT       ZBARX =  0.0          FT

                                           SYSTEMS CONTROLS
  IFCRW =           0 (NONE)     IFCRC =           0 (NONE)       IFC =           1 (NONE)       IDUMP =           0 (NONE)
  ICONV =         -10 (NONE)     ICNCL =           0 (NONE)
                                           COMPUTED QUANTITIES
 XBARTI =  0.0        FT        YBARTI =  0.0          FT      ZBARTI =  0.0          FT        YBARXI =  0.0          FT
 ZBARXI =  0.0        FT           DXT =  0.0          FT         DYT =  0.0          FT           DZT =  0.0          FT
    DYX =  0.0        FT           DZX =  0.0          FT        DVXT =  0.0          FT/SEC      DVYT =  0.0          FT/SEC
   DVZT =  0.0        FT/SEC      DVXX =  0.0          FT/SEC    DVYX =  0.0          FT/SEC      DVZX =  0.0          FT/SEC
     DT =  0.0        SEC         RADT =  0.0          FT        RADX =  0.0          FT          RHOT =  0.0          FT
   RHOX =  0.0        FT          DXTI =  0.0          FT        DYTI =  0.0          FT          DZTI =  0.0          FT
   DXTE =  0.0        FT          DYTE =  0.0          FT        DZTE =  0.0          FT          DYXI =  0.0          FT
   DZXI =  0.0        FT          DYXE =  0.0          FT        DZXE =  0.0          FT        DXDYT =  0.0          SQ-FT
  DYDZT =  0.0        SQ-FT      DYDZT =  0.0          SQ-FT    DYDZX =  0.0          SQ-FT     DXDYTI =  0.0          SQ-FT
 DXDZTI =  0.0        SQ-FT     DYDZTI =  0.0          SQ-FT   DYDZXI =  0.0          SQ-FT     DXDYTE =  0.0          SQ-FT
 DXDZTE =  0.0        SQ-FT     DYDZTE =  0.0          SQ-FT   DYDZXE =  0.0          SQ-FT
  ALPHA =  0.0        RAD         ATOT =  0.0          RAD     ALPTRM =  0.0          RAD       ALPHDO =  0.0          RAD/SC
      B =  1.00000 00 (NONE)    BETTRM =  0.0          RAD      BETAO =  0.0          RAD       BETADO =  0.0          RAD/SC
   BETA =  0.0        RAD       ALPMAX =  0.0          RAD     ALPMIN =  0.0          RAD         CAYO =  1.34990 06 SQ-F/S
    CDR =  3.10370-02 (NONE)       CMA = -6.07870-01 1/RAD     CMTHTD = -2.13070-05 SEC         CMTHTA =  0.0          SEC
   CAY1 =  0.0        RAD         CAY2 =  0.0          RAD      CCAOB =  4.21930-02 (NONE)        CDAB =  4.21930-02 (NONE)
  DELTT =  0.0        SEC         DELV =  0.0          FT/SEC    DELT =  0.0          SEC          DELX =  0.0          FT
  DYDXO =  0.0        RAD        DZDXO =  0.0          RAD      DYDTO =  0.0          FT/SEC     DZDTO =  0.0          FT/SEC
   DELW =  3.88300 02 1/SEC     DELLAM = -3.05380-02 1/SEC       EYEP =  7.?4970-07 SQ-SEC         EMP =  8.52040-02 SEC
   EDLT =  8.35200-11 (NONE)      EDLT =  9.72620-01 (NONE)         F =  1.?6700-03 1/FT             H =  2.55200 05 FT/SC2
  PSIDO =  0.0        RAD/SC      PHIO = -1.59940 00 RAD         PHI1 = -1.59910 00 RAD           PHI2 = -1.59960 00 RAD
   PSI? =  0.0        RAD           R1 =  0.0          RAD         R2 =  0.0          RAD             R3 =  0.0          RAD
     R4 =  0.0        RAD        RTRIM =  1.23330 00 (NONE)        TC =  9.09090-08 SEC            TCO =  9.09090-08 SEC
     TS =  1.18170 00 SEC          TGO =  0.0          SEC         TP =  9.09090-01 SEC         THETAO =  0.0          RAD
 THETDO =  0.0        RAD/SC       TOL =  1.00000-10 (NONE)        VC =  1.10000 04 FT/SEC         VCO =  1.16000 04 FT/SEC
    VYA =  0.0        FT/SEC       VZA =  0.0          FT/SEC      WX =  0.0          FT/SEC          WZ =  0.0          FT/SEC
     W3 =  8.01500 02 1/SEC         W1 =  5.03200 02 1/SEC         W2 =  1.27980 03 1/SEC          WL2 =  1.58960 06 1/SQ-S
  XLAM3 = -2.55270 01 1/SEC      XLAM1 = -2.55570 01 1/SEC      XLAM2 = -2.54960 01 1/SEC         XNU1 =  0.0          RAD
   XNU2 =  0.0        RAD           XG =  0.0          FT         XNU =  2.57270-02 (NONE)        XJAY =  0.0          RAD
   XJAZ =  0.0        RAD          XJA =  0.0          RAD         YA =  0.0          FT              ZA =  0.0          FT
  ALPHO =  0.0        RAD
     TN =  1.18170 00 SEC         XTFC =  1.00000 04 FT         YTFC =  0.0          FT           ZTFC =  0.0          FT
  VXTFC =  5.42590 03 FT/SEC     VYTFC =  0.0          FT/SEC   VZTFC =  0.0          FT/SEC          XT =  0.0          FT
     VT =  0.0        FT            ZT =  0.0          FT         VXT =  0.0          FT/SEC         VYT =  0.0          FT/SEC
    VZT =  0.0        FT/SEC        YR =  0.0          FT          ZR =  0.0          FT             VXR =  0.0          FT/SEC
    VYR =  0.0        FT/SEC       VZR =  0.0          FT/SEC      TX =  0.0          SEC
```

COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

Figure 4-6k Analytical Statistical Mode Check Problem Output

COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

```
                                  *** PROJECTILE DISPERSION CASE SUMMARY ***

                                           FIRE CONTROL PARAMETERS
     V0 =  1.1000D 04 FT/SEC      WX =  0.0          FT/SEC       WZ =  0.0          FT/SEC      RHO =  2.3780D-03 SL/CFT
    CX1 =  3.5854D-02 (NONE)      V1 =  1.6740D 04 FT/SEC        CX2 =  1.1951D-01 (NONE)        V2 =  3.9066D 03 FT/SEC
     XN =  1.0000D 04 FT         CNA =  1.9767D 00 1/RAD          SM =  6.2000D-02 100/L        CM0 = -7.5000D 00 (NONE)
   CMPA =  0.0        (NONE)   ATRMS =  0.0          DEG       PTRMS =  0.0          DEG      THETC =  0.0          DEG
   PSI0 =  0.0        DEG      TDOTO =  0.0          RAD/SC    PDOTO =  0.0          RAD/SC    GAMC =  0.0          DEG
    AZ0 =  0.0        DEG          A =  3.0680D-03 SQ-FT         ELL =  3.1000D-01 FT             D =  6.2500D-02 FT
      W =  1.1000D-01 LBS        AIX =  1.2346D-06 SL-SFT        AIY =  2.1103D-05 SL-SFT         P =  4.0000D 02 RAD/SC
 ALPCON =  5.0000D-01 DEG        CAA =  0.0          1/SDEG

                                           REAL WORLD PARAMETERS
     V0 =  1.1000D 04 FT/SEC      WX =  0.0          FT/SEC       WZ =  0.0          FT/SEC      RHO =  2.3780D-03 SL/CFT
    CX1 =  3.5800D-02 (NONE)      V1 =  1.6740D 04 FT/SEC        CX2 =  1.4340D-01 (NONE)        V2 =  3.9066D 03 FT/SEC
     XN =  1.0000D 04 FT         CNA =  1.9767D 00 1/RAD          SM =  6.0000D-02 100/L        CM0 = -7.5000D 00 (NONE)
   CMPA =  0.0        (NONE)   ATRMS =  0.0          DEG       PTRMS =  0.0          DEG      THETO =  0.0          DEG
   PSI0 =  0.0        DEG      TDOTO =  0.0          RAD/SC    PDOTO =  0.0          RAD/SC    GAMC =  0.0          DEG
    AZ0 =  0.0        DEG          A =  3.0680D-03 SQ-FT         ELL =  3.6000D-01 FT             D =  6.2500D-02 FT
      W =  1.9000D-01 LBS        AIX =  1.2346D-06 SL-SFT        AIY =  2.1103D-05 SL-SFT         P =  4.0000D 02 RAD/SC
 ALPCON =  5.0000D-01 DEG        CAA =  0.0          1/SDEG    XBART =  0.0          FT       YBART =  0.0          FT
  ZBART =  0.0        FT       YBARX =  0.0          FT        ZBARX =  0.0          FT

                                           SYSTEMS CONTROLS
   [illegible] =            0 (NONE)   ISCPC =            0 (NONE)       ISC =            0 (NONE)   IDUMP =            0 (NONE)
   [illegible] =          -10 (NONE)   ICNCL =            0 (NONE)
                                           COMPUTED QUANTITIES
 YBARTI =  0.0        FT      YBARTI =  0.0          FT       ZBARTI =  0.0          FT      YBARXI =  0.0          FT
 ZBARTI =  0.0        FT         DXT =  9.1023D 02 FT            DYT =  0.0          FT          DZT =  0.0          FT
    DYX =  0.0        FT         DZX =  0.0          FT         DVXT =  1.2685D 03 FT/SEC       DVYT =  0.0          FT/SEC
   DVZT =  0.0        FT/SEC    DVXX =  1.5376D 03 FT/SEC       DVYX =  0.0          FT/SEC     DVZX =  0.0          FT/SEC
     DT = -1.1623D-01 SEC       RADT =  9.1023D 02 FT           RADX =  0.0          FT         RHOT =  9.1023D 02 FT
   RHOX =  0.0        FT        DXTI =  9.1023D 02 FT           DYTI =  0.0          FT         DZTI =  0.0          FT
   DXTF =  9.1023D 02 FT        DYTF =  0.0          FT         DZTF =  0.0          FT         DYXI =  0.0          FT
   DZXI =  0.0        FT        DYXF =  0.0          FT         DZXF =  0.0          FT       DXDYT =  0.0          SQ-FT
  DYDZT =  0.0        SQ-FT    DYDZT =  0.0          SQ-FT     DYDZX =  0.0          SQ-FT   DXDYTI =  0.0          SQ-FT
 DXDYTI =  0.0        SQ-FT   DYDZTI =  0.0          SQ-FT    DYDZXI =  0.0          SQ-FT   DXDYTF =  0.0          SQ-FT
 DXDYTF =  0.0        SQ-FT   DYDZTF =  0.0          SQ-FT    DYDZXF =  0.0          SQ-FT
  ALPHA =  0.0        RAD       ATOT =  0.0          RAD      ALPTRM =  0.0          RAD     ALPHDO =  0.0          RAD/SC
      D =  1.0000D 05 (NONE)  BETTRM =  0.0          RAD       BETAO =  0.0          RAD     BETADO =  0.0          RAD/SC
  DELTA =  0.0        RAD     ALPMAX =  0.0          RAD      ALPMIN =  0.0          RAD       CAYO =  1.7352D 06 SQ-F/S
    CDA =  2.9604D-02 (NONE)     CMA = -5.6929D-01 1/RAD      CMTHTD = -2.1307D-05 SEC        CMTHTA =  0.0          SEC
   CAYI =  0.0        RAD       CAY2 =  0.0          RAD       CDASO =  4.3953D-02 (NONE)       CDAB =  4.3953D-02 (NONE)
  DELTT =  0.0        SEC       DELV =  0.0          FT/SEC     DELT =  0.0          SEC        DELX =  0.0          FT
  DYDXO =  0.0        RAD      DZDXO =  0.0          RAD       DYDTO =  0.0          FT/SEC    DZDTO =  0.0          FT/SEC
   DELY =  3.3370D 02 1/SEC   DELLAM = -9.2738D-02 1/SEC        EYEP =  7.6497D-07 SQ-SEC        GMP =  1.3942D-01 SEC
  [illegible] = 5.2448D-02 (NONE)  FOLT = 9.1415D-01 (NONE)       F =  6.0227D-04 1/FT            H =  1.5025D 05 FT/SC2
  PSIDO =  0.0        RAD/SC    PHIO = -1.5952D 00 RAD          PHI1 = -1.5942D 00 RAD          PHI2 = -1.5534D 00 RAD
   PSIO =  0.0        RAD         RI =  0.0          RAD          R2 =  0.0          RAD           R3 =  0.0          RAD
     ZA =  0.0        RAD      RTRIM =  1.2533D 00 (NONE)         TC =  9.0909D-08 SEC           TCO =  9.0909D-08 SEC
     TS =  1.0655D 00 SEC        TGO =  0.0          SEC          TP =  9.3909D-01 SEC        THETAO =  0.0          RAD
 THETOD =  0.0        RAD/SC     TOL =  1.0000D-10 (NONE)         VC =  1.1000D 04 FT/SEC        VCO =  1.1000D 04 FT/SEC
    VYA =  0.0        FT/SEC     VZA =  0.0          FT/SEC       WX =  0.0          FT/SEC       WZ =  0.0          FT/SEC
     WO =  4.6272D 02 1/SEC       WI =  4.7443D 02 1/SEC          W2 =  1.2510D 03 1/SEC         WL2 =  1.4986D 06 1/SQ-S
  XLAMO = -2.1015D 01 1/SEC    XLAMI = -2.1108D 01 1/SEC       XLAM2 = -2.0973D 01 1/SEC        XNU1 =  0.0          RAD
   XNU2 =  0.0        RAD         XG =  0.0          FT          XMU =  2.4611D-02 (NONE)       XJAY =  0.0          RAD
   XJAZ =  0.0        RAD        XJA =  0.0          RAD          YA =  0.0          FT            ZA =  0.0          FT
  ALPHO =  0.0        RAD
     TN =  1.1817D 00 SEC       XTFC =  1.0000D 04 FT           YTFC =  0.0          FT          ZTFC =  0.0          FT
  VXTFC =  6.4258D 03 FT/SEC   VYTFC =  0.0          FT/SEC    VZTFC =  0.0          FT/SEC         XT =  1.0012D 04 FT
     YT =  0.0        FT          ZT =  0.0          FT          VXT =  7.5973D 03 FT/SEC         VYT =  0.0          FT/SEC
    VZT =  0.0        FT/SEC      YR =  0.0          FT           ZR =  0.0          FT           VXR =  7.9665D 03 FT/SEC
    VYR =  0.0        FT/SEC     VZR =  0.0          FT/SEC       TX =  1.0655D 00 SEC
```

Figure 4-61 Analytical Statistical Mode Check Problem Output

```
*** NOTICE ***
    THE SPECIFICATION OF EITHER RHOT OR RHOX (IE
VARIABLES 320 OR 321) OR BOTH AS OUTPUT VARIABLES
NECESSITATES UP-DATING THE NOMINAL VALUES, DERIVATIVES,
MEANS, AND VARIANCES. THE LIMITING CASES FOR AND PARTIAL
DERIVATIVES OF RHOT AND RHOX ARE MEANINGLESS. THUS THE
FULL PROJECTILE DISPERSION CODE SHOULD BE UTILIZED IN
MONTE CARLO SIMULATIONS (MCALC = 0) TO OBTAIN
MEANINGFULL VALUES OF RHOT AND/OR RHOX. IF A TAYLOR
SERIES APPROXIMATION OF THE PROJECTILE DISPERSION CODE
(MCALC = 1, 2, OR 3) IS USED IN MONTE CARLO
SIMULATIONS THE RHOT AND RHOX RESULTS ARE MEANINGLESS.
```


Figure 4-6m Analytical Statistical Mode Check Problem Output

```
NOMINAL CASE, INDEPENDENT VARIABLES ...

     DE( 116)=  0.0          DE( 117)=  0.0          DE( 123)=  3.6000D-01    DE( 125)=  1.8300D-01
```

Figure 4-6n Analytical Statistical Mode Check Problem Output

```
NOMINAL CASE, DEPENDENT VARIABLES ...

   DE( 320)     DE( 321)     DE( 306)     DE( 307)     DE( 308)     DE( 309)     DE( 310)
   5.94770 00   1.66730-16   9.10230 02   0.0          0.0          0.0          0.0
```

Figure 4-6o Analytical Statistical Mode Check Problem Output

DERIVATIVES

DEPENDENT VARIABLE

IND. VAR.		320	321	306	307	308	309	310
115	1ST	0.0	0.0	0.0	1.0464D-01	-8.4757D-02	9.6250D-02	-7.7717D-02
	2ND	0.0	0.0	-9.9697D-01	4.0091D-17	0.0	3.7007D-17	0.0
117	1ST	0.0	0.0	0.0	8.4757D-02	1.0464D-01	7.7717D-02	9.6250D-02
	2ND	0.0	0.0	-9.9697D-01	3.0946D-18	3.5925D-15	4.5259D-15	3.2998D-16
123	1ST	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	2ND	0.0	0.0	0.0	0.0	0.0	0.0	0.0
125	1ST	0.0	0.0	1.0299D 04	0.0	0.0	0.0	0.0
	2ND	0.0	0.0	-1.0024D 05	0.0	0.0	0.0	0.0

Figure 4-6p Analytical Statistical Mode Check Problem Output

SECOND ORDER MIXED PARTIALS

IND. VAR.

					DEPENDENT VARIABLES			
		320	321	306	307	308	309	310
115	117	0.0	0.0	0.0	1.1440D-16	-1.8504D-17	1.0370D-16	-1.5420D-17
115	123	0.0	0.0	0.0	-2.9119D-01	2.3554D-01	-2.6777D-01	2.1599D-01
115	125	0.0	0.0	0.0	-4.9032D-01	3.9394D-01	-5.3831D-01	4.3426D-01
117	123	0.0	0.0	0.0	-2.3560D-01	-2.9119D-01	-2.1599D-01	-2.6777D-01
117	125	0.0	0.0	0.0	-3.9394D-01	-4.9032D-01	-4.3426D-01	-5.3831D-01
123	125	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 4-6q Analytical Statistical Mode Check Problem Output

2ND ORDER MEANS AND 1ST ORDER VARIANCES

	DEPENDENT VARIABLES						
	320	321	306	307	308	309	310
MEANS	5.4589D 01	1.5505D-01	9.0428D 02	2.1588D-17	1.7954D-15	2.0817D-17	1.6499D-16
VARIANCES	5.3073D 02	6.5585D-03	1.0477D 04	1.8133D-02	1.8133D-02	1.5304D-02	1.5304D-02
STD. DEV.	2.3038D 01	8.1046D-02	1.0236D 02	1.3466D-01	1.3466D-01	1.2371D-01	1.2371D-01

STANDARD DEVIATIONS WITH HIGHER PRECISION

2.3038D 01	2.3037507522147950D 01
8.1046D-02	8.1046232214[illegible]7000 02
1.0236D 02	1.0235684842431840D 02
1.3466D-01	1.3465729756933410D-01
1.3466D-01	1.3465729756933350D-01
1.2371D-01	1.2370910123153840D-01
1.2371D-01	1.2370910123153790D-01

Figure 4-6r Analytical Statistical Mode Check Problem Output

LIMITING CASES

INDEPENDENT VARIABLE			DEPENDENT VARIABLE							
NAME	VALUE	TOLERANCE	OE(306)	OE(207)	OE(308)	OE(309)	OE(310)	OE(311)	OE(312)	OE(313)
			OE(315)	OE(216)						
123	3.1000D-01	1.7900D-03	-9.0940D-12	0.0	0.0	0.0	0.0	2.7285D-12	0.0	0.0
			0.0	0.0						
			-9.0940D-12	0.0	0.0	0.0	0.0	2.7285D-12	0.0	0.0
			0.0	0.0						

NOTE:. This frame illustrates only the format of the LIMITING CASE printout. It was not generated by the input of Figure 4-5.

Figure 4-600 Analytical Statistical Mode Check Problem Output

4.4 Monte Carlo Mode

This problem demonstrates the Monte Carlo evaluation of dispersion statistics and probability distributions. The Monte Carlo mode determines dispersion by performing numerical experiments with the full non-linear trajectory equations rather than the Taylor series approximation used by the Analytical Statistical mode. Thus it is inherently more accurate, although large numbers of experiments would have to be conducted to realize the advantage. The number of experiments impacts cost. This mode is of interest to projectile designers and those interested in projectile dispersion statistics.

Figure 4-7 illustrates the input deck for a Monte Carlo mode simulation. The first card is the Group 1 card defined by Table 3-1. The Group 1 card sets MCØPT = 1, which forces the Statistical Processor into the Monte Carlo mode after using the results of the Analytical Statistical mode to set up the histogram boundaries. With the exception of the first card, all cards shown in Figure 4-7 belong to Group 2. Only TYPE = 2, 3, 4, 5, 7, and 8 cards may appear in a Monte Carlo simulation. There must be a minimum of one TYPE = 2, 3, or 4 and one TYPE = 7 or 8. Figure 4-7 illustrates the use of System Control IFCRW (CØDE# = 201) to initialize the Fire Control parameters at the Real World nominals.

Figure 4-8 illustrates the printed output generated by a Monte Carlo mode simulation. Frames 'a' through 'p' are the results of the initial Analytical Statistical mode calculations. The user is referred to Section 4.3 for a discussion of this output. Subsequent to the Analytical Statistical calculations, INTERNAL RANGE histograms are set up for all TYPE = 7 and TYPE = 8 variables with the cell boundaries determined by the analytically computed mean values and standard deviations. USER RANGE histograms are also set up for all variables with the cell boundaries for the TYPE = 8 variables determined by the information supplied on the Group 2 cards, as described in Table 3-2. At this point the Monte Carlo experiments begin.

Figure 4-8, Frame 'q' and Frame 'r' illustrate the summary information printed for each of the Monte Carlo experiments. This sequence of experiments is generated by using a random number generator to simulate random variations in the projectile parameters consistent with the input statistics. A given random sequence may be repeated by reusing the value of IRANNØ on the Group 1 card (see Table 3-1 for instructions). A different sequence will result from every selection of IRANNØ. The CASE variable in Frame 'q' counts the experiments. The first line, labeled IND. VAR., are the stochastic independent variable values appearing in the order given by Frame 'd'. The line labeled DEP. VAR. are the corresponding values of the dependent variables arranged in the order of Frame 'e'. Even though only a small number of experiments (100) were conducted, the summary print required ten pages (only two are included here). Thus, the user should consider suppressing the summary print by setting MCPRNT = 0 in Card Group 1, whenever large numbers of experiments are conducted. The information in the summary print is used to update the histograms after each experiment but is not stored in any other way.

At the conclusion of the Monte Carlo experiments, the histogram information is printed. In general, there are two sets of histograms. The INTERNAL RANGE histograms are printed first and are usually followed by the USER RANGE histograms. Since the example problem input shown in Figure 4-7 contains no TYPE = 8 variables, the sample output of Figure 4-8 contains only INTERNAL RANGE histograms. There is no need to document the USER RANGE histograms separately because they are constructed in an identical fashion and are subject to the same interpretation. The only difference is in how the cell boundaries are set up, which has already been discussed. The histogram data for the example problem commences in Frame 's' of Figure 4-8. The first line of print defines (1) the total number of random experiments performed (SAMPLE), (2) the number of experiments which were rejected from the INTERNAL RANGE histogram set because at least one of the dependent variable histogram ranges were exceeded (INTERNAL REJECTS) and (3) the number of experiments rejected from the USER RANGE histogram set for the same reason. The latter is zero in the example because USER RANGE histograms were never set up. The histograms making up the INTERNAL RANGE histogram set are presented on a variable-by-variable basis in Frames 's' through 'y'. Independent variable histograms precede the dependent variable histograms. Generally, the block of data describing the variable, consists of (1) a line identifying the variable and statistics computed from the histogram, (2) a second line defining the number of experiments rejected from the set because of range limitations on this variable and Analytical Statistical mode statistics, and (3) multiple lines of print defining the histogram on a cell-by-cell basis. The columns define the cell boundaries, the number of occurrences (FREQ), and the probability density function (DIST. FUNCT.). Although the format of the block is the same for all, there are some differences in interpretation. Table 4-1 rigorously interprets each printed value. Hash marks delimit the extent of the comments contained in the right hand column of Table 4-1.

```
1   6  0      1      0    20   100    50    50     112001
   206 5  0.0000000D00
   116 2  0.0000000D+0 3.0000000D+0 1.0000000D+0
   117 2  0.0000000D+0 3.0000000D+0 1.0000000D+0
   118 2  0.0000000D+0 2.7000000D+1 9.0000000D+1
   119 2  0.0000000D+0 2.7000000D+1 9.0000000D+1
   128 5  0.0000000D+0
   201 5  1.0000000D+0
   109 5  1.1730000D+3
   205 5 -8.0000000D+0
   306 7
   307 7
   308 7
   320 7
   317 7
   322 7
   323 7
   324 7
   518 7
   510 7
    -1
```

Figure 4-7 Monte Carlo Mode Check Problem Input

```
HYPERVELOCITY          INFLIGHT          TRAJECTORY          SCATTER

PREPARED BY                                   PREPARED FOR

     AVCO SYSTEMS DIVISION                         RESEARCH DIRECTORATE
     201 LOWELL STREET                             GENERAL THOMAS J. RODMAN LABORATORY
     WILMINGTON, MASSACHUSETTS 01887               ROCK ISLAND ARSENAL
                                                   ROCK ISLAND, ILLINOIS 61201

                      EFFECTIVE DATE - JANUARY 1976
```

Figure 4-8a Monte Carlo Mode Check Problem Output

```
                    *** SIMULATION INPUT SUMMARY ***

        SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

                IY =     1     IOPRNT =     6     ISPRNT =     0

                         MONTE CARLO CONTROLS
       MCOPT =     1     MCALC =     0     NCELL =    20     NTRIAL =   100
        IRJ1 =    50      IRJ2 =    50    MCPRNT =     1     IRANNO = 12001
      IOPLOT =     0    ISPLOT =     0
```

Figure 4-8b. Monte Carlo Mode Check Problem Output

***. INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
116	2	0.0	3.00000 00	1.00000 00	0
117	2	0.0	3.00000 00	1.00000 00	0
118	2	0.0	2.70000 01	9.00000 01	0
119	2	0.0	2.70000 01	9.00000 01	0
128	5	0.0	0.0	0.0	0
201	5	1.00000 00	0.0	0.0	0
109	5	1.17300 03	0.0	0.0	0
205	5	-8.00000 00	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
320	7	0.0	0.0	0.0	0
317	7	0.0	0.0	0.0	0
322	1	0.0	0.0	0.0	0
323		0.0	0.0	0.0	0
324		0.0	0.0	0.0	0
518	7	0.0	0.0	0.0	0
510	2	0.0	0.0	0.0	0

Figure 4-8c Monte Carlo Mode Check Problem Output

```
*** STOCHASTIC INDEPENDENT VARIABLES ***

THETO REAL WORLD          CODE NUMBER 116
PSIO REAL WORLD           CODE NUMBER 117
TDOTO REAL WORLD          CODE NUMBER 118
PDOTO REAL WORLD          CODE NUMBER 119
```

Figure 4-8d Monte Carlo Mode Check Problem Output

```
*** DEPENDENT VARIABLES ***

DXT  STATISTICAL VARIABLE    CODE NUMBER 306
DYT  STATISTICAL VARIABLE    CODE NUMBER 307
DZT  STATISTICAL VARIABLE    CODE NUMBER 308
RHOT STATISTICAL VARIABLE    CODE NUMBER 320
DT   STATISTICAL VARIABLE    CODE NUMBER 317
DXTI STATISTICAL VARIABLE    CODE NUMBER 322
DYTI STATISTICAL VARIABLE    CODE NUMBER 323
DZTI STATISTICAL VARIABLE    CODE NUMBER 324
TX   TRAJECTORY VARIABLE     CODE NUMBER 518
VXT  TRAJECTORY VARIABLE     CODE NUMBER 510
```

Figure 4-8e Monte Carlo Mode Check Problem Output

```
*** VARIABLES RESET BY INPUT SEQUENCE ***

   XN REAL WORLD          CODE NUMBER 109
    P REAL WORLD          CODE NUMBER 128
IFCRW SYSTEM CONTROL      CODE NUMBER 201
ICONV SYSTEM CONTROL      CODE NUMBER 205
ICNCL SYSTEM CONTROL      CODE NUMBER 206
```

Figure 4-8f Monte Carlo Mode Check Problem Output

```
*** VARIABLES ASSIGNED PRESET VALUES ***

     CX1 FIRE CONTROL        CODE NUMBER   5
      V1 FIRE CONTROL        CODE NUMBER   6
     CX2 FIRE CONTROL        CODE NUMBER   7
      V2 FIRE CONTROL        CODE NUMBER   8
      XN FIRE CONTROL        CODE NUMBER   9
     CNA FIRE CONTROL        CODE NUMBER  10
      SM FIRE CONTROL        CODE NUMBER  11
     CMQ FIRE CONTROL        CODE NUMBER  12
    CMPA FIRE CONTROL        CODE NUMBER  13
   ATRMS FIRE CONTROL        CODE NUMBER  14
   BTRMS FIRE CONTROL        CODE NUMBER  15
   THETO FIRE CONTROL        CODE NUMBER  16
    PSIO FIRE CONTROL        CODE NUMBER  17
   TDOTO FIRE CONTROL        CODE NUMBER  18
   PDOTO FIRE CONTROL        CODE NUMBER  19
    GAMO FIRE CONTROL        CODE NUMBER  20
     A70 FIRE CONTROL        CODE NUMBER  21
       A FIRE CONTROL        CODE NUMBER  22
     ELL FIRE CONTROL        CODE NUMBER  23
       D FIRE CONTROL        CODE NUMBER  24
       W FIRE CONTROL        CODE NUMBER  25
     AIX FIRE CONTROL        CODE NUMBER  26
     AIY FIRE CONTROL        CODE NUMBER  27
       P FIRE CONTROL        CODE NUMBER  28
  ALPCON FIRE CONTROL        CODE NUMBER  29
     CAA FIRE CONTROL        CODE NUMBER  30
      VO REAL WORLD          CODE NUMBER 101
      WX REAL WORLD          CODE NUMBER 102
      WZ REAL WORLD          CODE NUMBER 103
     RHO REAL WORLD          CODE NUMBER 104
     CX1 REAL WORLD          CODE NUMBER 105
      V1 REAL WORLD          CODE NUMBER 106
     CX2 REAL WORLD          CODE NUMBER 107
      V2 REAL WORLD          CODE NUMBER 108
     CNA REAL WORLD          CODE NUMBER 110
      SM REAL WORLD          CODE NUMBER 111
     CMQ REAL WORLD          CODE NUMBER 112
    CMPA REAL WORLD          CODE NUMBER 113
   ATRMS REAL WORLD          CODE NUMBER 114
   BTRMS REAL WORLD          CODE NUMBER 115
      VO FIRE CONTROL        CODE NUMBER   1
      WX FIRE CONTROL        CODE NUMBER   2
      WZ FIRE CONTROL        CODE NUMBER   3
     RHO FIRE CONTROL        CODE NUMBER   4
    GAMO REAL WORLD          CODE NUMBER 120
     A70 REAL WORLD          CODE NUMBER 121
       A REAL WORLD          CODE NUMBER 122
     ELL REAL WORLD          CODE NUMBER 123
       D REAL WORLD          CODE NUMBER 124
       W REAL WORLD          CODE NUMBER 125
     AIX REAL WORLD          CODE NUMBER 126
     AIY REAL WORLD          CODE NUMBER 127
  ALPCON REAL WORLD          CODE NUMBER 129
     CAA REAL WORLD          CODE NUMBER 130
   XBART REAL WORLD          CODE NUMBER 131
   YBART REAL WORLD          CODE NUMBER 132
   ZBART REAL WORLD          CODE NUMBER 133
   YBARX REAL WORLD          CODE NUMBER 134
   ZBARX REAL WORLD          CODE NUMBER 135
   IFCRC SYSTEM CONTROL      CODE NUMBER 202
```

Figure 4-8g Monte Carlo Mode Check Problem Output

IDUMP SYSTEM CONTROL CODE NUMBER 204

Figure 4-8h Monte Carlo Mode Check Problem Output

```
                                   *** PROJECTILE DISPERSION CASE SUMMARY ***
                                             FIRE CONTROL PARAMETERS
     V0 =  1.1000D 04 FT/SEC        WX =  0.0         FT/SEC        WZ =  0.0         FT/SEC        RHO =  2.3780D-03 SL/CFT
    CX1 =  3.5854D-02 (NONE)        V1 =  1.6740D 04 FT/SEC        CX2 =  1.1951D-01 (NONE)         V2 =  3.9060D 03 FT/SFC
     XN =  1.1730D 03 FT           CNA =  1.9767D 00 1/RAD          SM =  6.2000D-02 100/L         CMQ = -7.5000D 00 (NONE)
   CMPA =  0.0        (NONE)     ATRMS =  0.0         DEG        BTRMS =  0.0         DEG        THETO =  0.0         DEG
   PSIO =  0.0        DEG        TDOTO =  0.0         RAD/SC     POOTO =  0.0         RAD/SC      GAMO =  0.0         DEG
    AZO =  0.0        DEG            A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT              D =  6.2500D-02 FT
      W =  1.1000D-01 LBS          AIX =  1.2346D-06 SL-SFT       AIY =  2.1103D-05 SL-SFT          P =  0.0         RAD/SC
 ALPCON =  5.0000D-01 DEG          CAA =  0.0         1/SDEG
                                              REAL WORLD PARAMETERS
     V0 =  1.1000D 04 FT/SEC        WX =  0.0         FT/SEC        WZ =  0.0         FT/SEC        RHO =  2.3780D-03 SL/CFT
    CX1 =  3.5854D-02 (NONE)        V1 =  1.6740D 04 FT/SEC        CX2 =  1.1951D-01 (NONE)         V2 =  3.9060D 03 FT/SEC
     XN =  1.1730D 03 FT           CNA =  1.9767D 00 1/RAD          SM =  6.2000D-02 100/L         CMQ = -7.5000D 00 (NONE)
   CMPA =  0.0        (NONE)     ATRMS =  0.0         DEG        BTRMS =  0.0         DEG        THETO =  0.0         DEG
   PSIO =  0.0        DEG        TDOTO =  0.0         RAD/SC     PDOTO =  0.0         RAD/SC      GAMO =  0.0         DEG
    AZO =  0.0        DEG            A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT              D =  6.2500D-02 FT
      W =  1.1000D-01 LBS          AIX =  1.2346D-06 SL-SFT       AIY =  2.1103D-05 SL-SFT          P =  0.0         RAD/SC
 ALPCON =  5.0000D-01 DEG          CAA =  0.0         1/SDEG     XBART =  0.0         FT         YBART =  0.0         FT
  ZBART =  0.0        FT         YBARX =  0.0         FT         ZBARX =  0.0         FT
                                                SYSTEMS CONTROLS
  IFCRW =           1 (NONE)     IFCRC =            0 (NONE)       IFC =            1 (NONE)     IDUMP =            0 (NONE)
  ICONV =          -8 (NONE)     ICNCL =            0 (NONE)
                                               COMPUTED QUANTITIES
 XBARTI =  0.0        FT        YBARTI =  0.0         FT        ZBARTI =  0.0         FT        YBARXI =  0.0         FT
 ZBARXI =  0.0        FT           DXT =  0.0         FT           DYT =  0.0         FT           DZT =  0.0         FT
    DYX =  0.0        FT           DZX =  0.0         FT          DVXT =  0.0         FT/SEC       DVYT =  0.0         FT/SEC
   DVZT =  0.0        FT/SEC      DVXX =  0.0         FT/SEC      DVYX =  0.0         FT/SEC       DVZX =  0.0         FT/SEC
     DT =  0.0        SEC         RADT =  0.0         FT          RADX =  0.0         FT           RHOT =  0.0         FT
   RHOX =  0.0        FT          DXTI =  0.0         FT          DYTI =  0.0         FT           DZTI =  0.0         FT
   DXTE =  0.0        FT          DYTE =  0.0         FT          DZTE =  0.0         FT           DYXI =  0.0         FT
   DZXI =  0.0        FT          DYXE =  0.0         FT          DZXE =  0.0         FT          DXDYT =  0.0         SQ-FT
  DXDZT =  0.0        SQ-FT      DYDZT =  0.0         SQ-FT      DYDZX =  0.0         SQ-FT      DXDYTI =  0.0         SQ-FT
 DXDZTI =  0.0        SQ-FT     DYDZTI =  0.0         SQ-FT     DYDZXI =  0.0         SQ-FT      DXDYTE =  0.0         SQ-FT
 DXDZTE =  0.0        SQ-FT     DYDZTE =  0.0         SQ-FT     DYDZXE =  0.0         SQ-FT
  ALPHA =  0.0        RAD         ATOT =  0.0         RAD       ALPTRM =  0.0         RAD        ALPHOO =  0.0         RAD/SC
      B =  1.0000D 00 (NONE)    BETTRM =  0.0         RAD        BETAO =  0.0         RAD        BETADO =  0.0         RAD/SC
   BETA =  0.0        RAD       ALPMAX =  0.0         RAD       ALPMIN =  0.0         RAD          CAYO =  1.3499D 06 SQ-F/S
    COR =  3.1037D-02 (NONE)       CMA = -6.0787D-01 1/RAD      CMTHTC = -2.1307D-05 SEC         CMTHTA =  0.0         SEC
   CAY1 =  0.0        RAD         CAY2 =  0.0         RAD        CDACB =  4.2193D-02 (NONE)        CDAB =  4.2193D-02 (NONE)
  DELTT =  0.0        SEC         DELV =  0.0         FT/SEC      DELT =  0.0         SEC          DELX =  0.0         FT
  DYDXO =  0.0        RAD        DZDXO =  0.0         RAD        DYDTO =  0.0         FT/SEC      DZDTO =  0.0         FT/SEC
   DELW =  0.0        1/SEC     DELLAM =  0.0         1/SEC       EYEF =  7.6497D-07 SQ-SEC         EYP =  8.5204D-02 SEC
   EDLT =  6.5739D-02 (NONE)      SDLT =  1.0000D 00 (NONE)          F =  1.0670D-03 1/FT             H =  2.5520D 05 FT/SC2
  PSIDO =  0.0        RAD/SC      PHIO = -1.5994D 00 RAD          PHI1 = -1.5994D 00 RAD          PHI2 = -1.5994D 00 RAD
   PSIO =  0.0        RAD           R1 =  0.0         RAD           R2 =  0.0         RAD            R3 =  0.0         RAD
     RA =  0.0        RAD        RTPIM =  1.0000D 00 (NONE)         TC =  9.0909D-08 SEC            TCO =  9.0909D-09 SEC
     TS =  1.0953D-01 SEC          TGO =  0.0         SEC           TP =  1.0664D-01 SEC         THETAO =  0.0         RAD
 THETDO =  0.0        RAD/SC       TOL =  1.0000D-05 (NONE)         VC =  1.1000D 04 FT/SEC         VCO =  1.1000D 04 FT/SEC
    VYA =  0.0        FT/SEC       VZA =  0.0         FT/SEC        WX =  0.0         FT/SEC         WZ =  0.0         FT/SEC
     WO =  8.9142D 02 1/SEC         W1 =  8.9142D 02 1/SEC          W2 =  8.9142D 02 1/SEC          WL2 =  1.5893D 05 1/SQ-S
  XLAMO = -2.5527D 01 1/SEC      XLAM1 = -2.5527D 01 1/SEC       XLAM2 = -2.5527D 01 1/SEC         XNU1 =  0.0         RAD
   XNU2 =  0.0        RAD           XG =  0.0         FT           XMU =  2.8727D-02 (NONE)        XJAY =  0.0         RAD
   XJAZ =  0.0        RAD          XJA =  0.0         RAD           YA =  0.0         FT             ZA =  0.0         FT
  ALPHO =  0.0        RAD
     TN =  1.0953D-01 SEC         XTFC =  1.1730D 03 FT           YTFC =  0.0         FT           ZTFC =  0.0         FT
  VXTFC =  1.0426D 04 FT/SEC     VYTFC =  0.0         FT/SEC     VZTFC =  0.0         FT/SEC         XT =  0.0         FT
     YT =  0.0        FT            ZT =  0.0         FT           VXT =  0.0         FT/SEC         VYT =  0.0         FT/SEC
    VZT =  0.0        FT/SEC        YR =  0.0         FT            ZR =  0.0         FT            VXR =  0.0         FT/SEC
    VYR =  0.0        FT/SEC       VZR =  0.0         FT/SEC        TX =  0.0         SEC
```

Figure 4-8i Monte Carlo Mode Check Problem Output

```
                              *** PROJECTILE DISPERSION CASE SUMMARY ***

                                       FIRE CONTROL PARAMETERS
    V0 =  1.1000D 04 FT/SEC      WX =  0.0          FT/SEC       WZ =  0.0          FT/SEC       RHO =  2.3780D-03 SL/CFT
   CX1 =  3.5854D-02 (NONE)      V1 =  1.6740D 04 FT/SEC      CX2 =  1.1951D-01 (NONE)        V2 =  3.9060D 03 FT/SEC
    XN =  1.1730D 03 FT         CNA =  1.9767D 00 1/RAD        SM =  6.2000D-02 100/L        CM0 = -7.5000D 00 (NCNE)
  CMPA =  0.0        (NONE)   ATRMS =  0.0          DEG       BTRMS =  0.0          DEG        THETO =  0.0          DEG
  PSI0 =  0.0        DEG      TOOTO =  0.0          RAD/SC    POOTO =  0.0          RAD/SC      GAM0 =  0.0          DEG
   AZ0 =  0.0        DEG          A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT             D =  6.2500D-02 FT
     W =  1.1000D-01 LBS        AIX =  1.2346D-06 SL-SFT       AIY =  2.1103D-05 SL-SFT          P =  0.0          RAD/SC
ALPCON =  5.0000D-01 DEG        CAA =  0.0          1/SDEG

                                       REAL WORLD PARAMETERS
    V0 =  1.1000D 04 FT/SEC      WX =  0.0          FT/SEC       WZ =  0.0          FT/SEC       RHO =  2.3780D-03 SL/CFT
   CX1 =  3.5854D-02 (NONE)      V1 =  1.6740D 04 FT/SEC      CX2 =  1.1951D-01 (NONE)        V2 =  3.9060D 03 FT/SEC
    XN =  1.1730D 03 FT         CNA =  1.9767D 00 1/RAD        SM =  6.2000D-02 100/L        CM0 = -7.5000D 00 (NCNE)
  CMPA =  0.0        (NONE)   ATRMS =  0.0          DEG       BTRMS =  0.0          DEG        THETO =  0.0          DEG
  PSI0 =  0.0        DEG      TOOTO =  0.0          RAD/SC    POOTO =  0.0          RAD/SC      GAM0 =  0.0          DEG
   AZ0 =  0.0        DEG          A =  3.0680D-03 SQ-FT        ELL =  3.1000D-01 FT             D =  6.2500D-02 FT
     W =  1.1000D-01 LBS        AIX =  1.2346D-06 SL-SFT       AIY =  2.1103D-05 SL-SFT          P =  0.0          RAD/SC
ALPCON =  5.0000D-01 DEG        CAA =  0.0          1/SDEG     XPART =  0.0          FT         YBART =  0.0          FT
 ZBART =  0.0        FT       YBARX =  0.0          FT         ZBARX =  0.0          FT

                                       SYSTEMS CONTROLS
 IFCRM =          1 (NONE)    IFCRC =            0 (NCNE)        IFC =            0 (NONE)     IDUMP =            0 (NCNE)
 ICONV =         -8 (NONE)    ICNCL =            0 (NCNE)
                                       COMPUTED QUANTITIES
XBARTI =  0.0        FT      YBARTI =  0.0          FT        ZBARTI =  0.0          FT       YBARXI =  0.0          FT
ZBARXI =  0.0        FT         DXT = -5.4001D-12 FT             DYT =  0.0          FT          DZT =  0.0          FT
   DYX =  0.0        FT         DZX =  0.0          FT            DVXT =  2.7285D-12 FT/SEC       DVYT =  0.0          FT/SEC
  DVZT =  0.0        FT/SEC    DVXX =  0.0          FT/SEC      DVYX =  0.0          FT/SEC       DVZX =  0.0          FT/SEC
    DT =  0.0        SEC       RADI =  5.4001D-12 FT            RADX =  0.0          FT           RHOT =  5.4001D-12
  RHOX =  0.0        FT         DXTI = -5.4001D-12 FT             DYTI =  0.0          FT          DZTI =  0.0
  DXTE = -5.4001D-12 FT         DYTE =  0.0          FT            DZTE =  0.0          FT          DYXI =  0.0          F
  DZXI =  0.0        FT         DYXE =  0.0          FT            DZXE =  0.0          FT         DXDYT =  0.0          SQ-FT
 DXDZT =  0.0        SQ-FT     DYDZT =  0.0          SQ-FT        DYDZX =  0.0          SQ-FT     DXDYTI =  0.0          SQ-FT
DXDZTI =  0.0        SQ-FT    DYDZTI =  0.0          SQ-FT       DYDZXI =  0.0          SQ-FT     DXDYTE =  0.0          SQ-FT
DXDZTE =  0.0        SQ-FT    DYDZTE =  0.0          SQ-FT       DYDZXE =  0.0          SQ-FT
 ALPHA =  0.0        RAD        ATOT =  0.0          RAD         ALFTRM =  0.0          RAD        ALPHDO =  0.0          RAD/SC
     G =  1.0000D 00 (NONE)   BETTRM =  0.0          RAD          BETA0 =  0.0          RAD        BETADO =  0.0          RAD/SC
  BETA =  0.0        RAD      ALPMAX =  0.0          RAD         ALPMIN =  0.0          RAD          CAYD =  1.3499D 06 SQ-F/S
   CD0 =  3.1037D-02 (NONE)      CMA = -6.0787D-01 1/RAD       CMTHTO = -2.1307D-05 SEC         CMTHTA =  0.0          SEC
  CAY1 =  0.0        RAD        CAY2 =  0.0          RAD          CDAOR =  4.2193D-02 (NONE)      CDAB =  4.2193D-02 (NONE)
 DELTT =  0.0        SEC        DELV =  0.0          FT/SEC        DELT =  0.0          SEC          DELX =  0.0          FT
 DYDX0 =  0.0        RAD       DZDX0 =  0.0          RAD          DYDT0 =  0.0          FT/SEC      DZDT0 =  0.0          FT/SEC
  DELW =  0.0        1/SEC    DELLAM =  0.0          1/SEC         EYEP =  7.6497D-07 SQ-SEC        EMP =  8.5204D-02 SEC
  FDLT =  6.5739D-02 (NONE)     EDLT =  1.0000D 00 (NCNE)             F =  1.0670D-03 1/FT            H =  2.5523D 05 FT/SC2
 PSIDO =  0.0        RAD/SC     PHI0 = -1.5994D 00 RAD            PHI1 = -1.5994D 00 RAD          PHI2 = -1.5994D 00 RAD
  PSI1 =  0.0        RAD          R1 =  0.0          RAD              R2 =  0.0          RAD            R3 =  0.0          RAD
    T4 =  0.0        RAD       RTRIM =  1.0000D 00 (NCNE)            TC =  9.0909D-08 SEC           TC0 =  9.0909D-08 SEC
    TS =  1.0953D-01 SEC         TG0 =  0.0          SEC              TP =  1.0654D-01 SFC        THETAO =  0.0          RAD
THETDO =  0.0        RAD/SC      TOL =  1.0000D-08 (NCNE)            VC =  1.1000D 04 FT/SEC         VC0 =  1.1000D 04 FT/SEC
   VYA =  0.0        FT/SEC      VZA =  0.0          FT/SEC          WX =  0.0          FT/SEC          WZ =  0.0          FT/SEC
    W0 =  8.9142D 02 1/SEC        W1 =  8.9142D 02 1/SEC            W2 =  8.9142D 02 1/SEC          WL2 =  1.5893D 06 1/SQ-S
 XLAM0 = -2.5527D 01 1/SEC     XLAM1 = -2.5527D 01 1/SEC         XLAM2 = -2.5527D 01 1/SEC         XNU1 =  0.0          RAD
  XNU2 =  0.0        RAD          XG =  0.0          FT             XMU =  2.8727D-02 (NONE)        XJAY =  0.0          RAD
  XJAZ =  0.0        RAD         XJA =  0.0          RAD             YA =  0.0          FT             ZA =  0.0          FT
 ALPH0 =  0.0        RAD
    TN =  1.0953D-01 SEC        XTFC =  1.1730D 03 FT             YTFC =  0.0          FT           ZTFC =  0.0          FT
 VXTFC =  1.0426D 04 FT/SEC    VYTFC =  0.0          FT/SEC        VZTFC =  0.0          FT/SEC          XT =  1.1730D 03 FT
    YT =  0.0        FT           ZT =  0.0          FT              VXT =  1.0426D 04 FT/SEC          VYT =  0.0          FT/SEC
   VZT =  0.0        FT/SEC       YR =  0.0          FT               ZR =  0.0          FT             VXR =  1.0426D 04 FT/SEC
   VYR =  0.0        FT/SEC      VZR =  0.0          FT/SEC           TX =  1.0953D-01 SEC
```

Figure 8-j Monte Carlo Mode Check Problem Output

```
*** NOTICE ***

     THE SPECIFICATION OF EITHER RHOT OR RHOX (IE
VARIABLES 320 OR 321) OR BOTH AS OUTPUT VARIABLES
NECESSITATES UP-DATING THE NOMINAL VALUES, DERIVATIVES,
MEANS, AND VARIANCES. THE LIMITING CASES FOR AND PARTIAL
DERIVATIVES OF RHOT AND RHOX ARE MEANINGLESS. THUS THE
FULL PROJECTILE DISPERSION CODE SHOULD BE UTILIZED IN
MONTE CARLO SIMULATIONS (MCALC = 0) TO OBTAIN
MEANINGFULL VALUES OF RHOT AND/OR RHOX. IF A TAYLOR
SERIES APPROXIMATION OF THE PROJECTILE DISPERSION CODE
(MCALC = 1, 2, OR 3) IS USED IN MONTE CARLO
SIMULATIONS THE RHOT AND RHOX RESULTS ARE MEANINGLESS.
```

Figure 4-8k Monte Carlo Mode Check Problem Output

```
NOMINAL CASE, INDEPENDENT VARIABLES ...
      OE( 116)=  0.0        OE( 117)=  0.0        OE( 118)=  0.0        OE( 119)=  0.0
```

Figure 4-81 Monte Carlo Mode Check Problem Output

NOMINAL CASE, DEPENDENT VARIABLES ...

DE(305)	DE(307)	DE(308)	DE(320)	DE(317)	DE(322)	DE(323)	DE(324)	DE(518)	DE(510)
-5.4001D-12	0.0	0.0	3.8934D 00	0.0	-5.4001D-12	0.0	0.0	1.0953D-01	1.0426D 04

Figure 4-8m Monte Carlo Mode Check Problem Output

DERIVATIVES

IND. VAR.		306	307	308	320	317	322	323	324	518	513
						DEPENDENT VARIABLE					
116	1ST	0.0	2.1939D-02	3.0679D-17	0.0	0.0	0.0	2.1939D-02	3.0679D-17	0.0	0.0
	2ND	-1.0417D-01	0.0	3.2544D-19	0.0	9.9972D-06	-1.0417D-01	0.0	3.2544D-19	9.9972D-06	-1.3579D 00
117	1ST	0.0	-3.0679D-17	2.1939D-02	0.0	0.0	0.0	-3.0679D-17	2.1939D-02	0.0	0.0
	2ND	-1.0417D-01	0.0	3.0840D-17	0.0	9.9972D-05	-1.0417D-01	0.0	3.0840D-17	9.9972D-06	-1.3579D 00
118	1ST	0.0	3.4177D-02	-3.9091D-21	0.0	0.0	0.0	3.4177D-02	-3.9091D-21	0.0	0.0
	2ND	-4.6780D-04	0.0	2.8957D-22	0.0	4.4877D-05	-4.6780D-04	0.0	2.8957D-22	4.4877D-08	-5.3385D-03
119	1ST	0.0	0.0	3.4177D-02	0.0	0.0	0.0	0.0	3.4177D-02	0.0	0.0
	2ND	-4.6780D-04	0.0	0.0	0.0	4.4877D-05	-4.6780D-04	0.0	0.0	4.4877D-08	-5.3385D-03

Figure 4-8n Monte Carlo Mode Check Problem Output

SECOND ORDER MIXED PARTIALS

IND. VAR.		DEPENDENT VARIABLES 306	307	308	320	317	322	323	324	518	510
115	117	0.0	2.6985D-17	4.8187D-19	0.0	0.0	0.0	2.6985D-17	4.8187D-19	0.0	0.0
115	118	-2.3849D-04	-1.3706D-18	2.9449D-19	0.0	2.2883D-08	-2.3849D-04	-1.3706D-18	2.9449D-19	2.2883D-08	-2.7458D-03
115	119	0.0	9.8515D-19	0.0	0.0	0.0	0.0	9.8515D-19	0.0	0.0	0.0
117	118	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
117	119	-2.3849D-04	0.0	2.7413D-18	0.0	2.2883D-08	-2.3849D-04	0.0	2.7413D-18	2.2883D-08	-2.7458D-03
118	119	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 4-8o Monte Carlo Mode Check Problem Output

2ND ORDER MEANS AND VARIANCES ASSUMING GAUSSIAN INDEPENDENT VARIABLES

	DEPENDENT VARIABLES									
	306	307	308	320	317	322	323	324	518	510
MEANS	-3.89340 00	0.0	1.67550-17	5.28870 00	3.73500-04	-3.89340 00	0.0	1.67550-17	1.09990-01	1.02820 04
VARIANCES	1.43770 01	9.46160 00	9.46160 00	4.98150 00	1.32240-07	1.43770 01	9.46160 00	9.46160 00	1.32240-07	1.87210 03
STD. DEV.	3.79080 00	3.07600 00	3.07600 00	2.23190 00	3.63650-04	3.79080 00	3.07600 00	3.07600 00	3.63650-04	4.32680 01

STANDARD DEVIATIONS WITH HIGHER PRECISION

3.79080 00	3.7907513799174500D 00
3.07600 00	3.0759670705402220D 00
3.07600 00	3.0759670705402220D 00
2.23190 00	2.2319327327391050D 00
3.63650-04	3.6365050821273700D-04
3.79080 00	3.7907513799174500D 00
3.07600 00	3.0759670705402220D 00
3.07600 00	3.0759670705402220D 00
3.63650-04	3.6365050821273700D-04
4.32680 01	4.3267501719467510D 01

Figure 4-8p Monte Carlo Mode Check Problem Output

*** SUMMARY OF MONTE CARLO RANDOM EXPERIMENTS ***

```
CASE      1
IND. VAR.  8.2460D-02 -5.3724D-01  2.6440D 02 -5.6835D 01
DEP. VAR. -1.1963D 01  8.9527D 00 -1.9357D 00  1.2207D 01   1.1595D-03 -8.0598D 00  8.9527D 00 -1.9357D 00  1.1069D-01  1.0217D 04

CASE      2
IND. VAR.  3.7097D-01 -5.6522D-02 -2.6124D 02  6.1215D 01
DEP. VAR. -1.1746D 01 -8.8370D 00  2.0715D 00  1.2002D 01   1.1383D-03 -7.8530D 00 -8.8370D 00  2.0715D 00  1.1067D-01  1.0221D 04

CASE      3
IND. VAR. -1.7235D-01  1.0733D 00 -6.5338D 01  7.3043D 01
DEP. VAR. -1.7356D 00 -2.2340D 00  2.5167D 00  3.9975D 00   1.6693D-04  2.1578D 00 -2.2340D 00  2.5167D 00  1.0970D-01  1.0397D 04

CASE      4
IND. VAR.  2.4335D-01 -1.9407D-02  8.9523D 00  8.4867D 01
DEP. VAR. -1.3330D 00  3.1100D-01  2.8973D 00  3.8790D 00   1.2811D-04  2.5604D 00  3.1100D-01  2.8973D 00  1.0966D-01  1.0405D 04

CASE      5
IND. VAR. -2.2644D 00  2.0103D 00 -9.9220D 01 -2.1140D 01
DEP. VAR. -2.1254D 00 -3.4308D 00 -6.7736D-01  3.9185D 00   2.0457D-04  1.7679D 00 -3.4308D 00 -6.7736D-01  1.0973D-01  1.0390D 04

CASE      6
IND. VAR. -1.2957D 00 -2.8326D 00 -1.1184D 02  2.3479D 01
DEP. VAR. -2.5581D 00 -3.8433D 00  7.3894D-01  4.1352D 00   2.4643D-04  1.3352D 00 -3.8433D 00  7.3894D-01  1.0978D-01  1.0382D 04

CASE      7
IND. VAR.  2.3230D-01 -1.2331D 00 -3.8477D 01  2.1260D 02
DEP. VAR. -7.6009D 00 -1.3021D 00  7.1956D 00  8.1586D 00   7.3879D-04 -3.7076D 00 -1.3021D 00  7.1956D 00  1.1027D-01  1.0292D 04

CASE      8
IND. VAR.  6.4429D-01  6.9085D-01  1.1397D 02  1.7362D 01
DEP. VAR. -2.2384D 00  3.9027D 00  6.0749D-01  4.2824D 00   2.1551D-04  1.6550D 00  3.9027D 00  6.0749D-01  1.0975D-01  1.0387D 04

CASE      9
IND. VAR. -4.1117D-02  4.6138D-01  4.7775D 01 -1.1122D 02
DEP. VAR. -2.3946D 00  1.6290D 00 -3.7843D 00  4.3841D 00   2.3064D-04  1.4987D 00  1.6290D 00 -3.7843D 00  1.0976D-01  1.0384D 04

CASE     10
IND. VAR. -1.6145D-01  9.2421D-02 -3.8744D 01 -8.7503D 01
DEP. VAR. -1.6131D 00 -1.3261D 00 -2.9850D 00  3.5836D 00   1.5511D-04  2.2803D 00 -1.3261D 00 -2.9850D 00  1.0969D-01  1.0400D 04
```

Figure 4-8q Monte Carlo Mode Check Problem Output

```
CASE    95
IND. VAR. -1.3260D 00 -5.3271D-01  1.4574D 02  7.1560D 01  4.1963D-04 -4.4873D-01  4.9351D 00  2.4258D 00  1.0995D-01  1.0350D 04
DEP. VAR. -4.3421D 00  4.9351D 00  2.4258D 00  5.5174D 00

CASE    96
IND. VAR. -1.2033D-01  3.4159D-01  6.3289D 01  8.4953D 01  1.8442D-04  1.9767D 00  2.1573D 00  2.9068D 00  1.0971D-01  1.0394D 04
DEP. VAR. -1.9167D 00  2.1573D 00  2.9068D 00  4.1244D 00

CASE    97
IND. VAR. -1.1267D 00 -1.2769D-01 -1.7649D 01 -2.4597D 00  1.4574D-05  3.7414D 00 -6.2791D-01 -8.6869D-02  1.0954D-01  1.0424D 04
DEP. VAR. -1.5193D-01 -6.2791D-01 -8.6869D-02  3.7948D 00

CASE    98
IND. VAR. -9.9821D-01  2.5412D-01 -1.0800D 02 -4.6585D 01  2.2326D-04  1.5750D 00 -3.7065D 00 -1.5838D 00  1.0975D-01  1.0305D 04
DEP. VAR. -2.3184D 00 -3.7065D 00 -1.5838D 00  4.3275D 00

CASE    99
IND. VAR.  5.9933D-01  1.2344D-01  8.9032D 01 -2.3536D 02  1.0031D-03 -5.3750D 00  3.0312D 00 -7.9758D 00  1.1053D-01  1.0245D 04
DEP. VAR. -1.0271D 01  3.0312D 00 -7.9758D 00  1.0653D 01

CASE   100
IND. VAR.  4.1290D-01  6.5879D-01  4.0767D 00 -9.3185D 00  4.7415D-05  3.8439D 00  1.4841D-01 -3.0404D-01  1.0954D-01  1.0426D 04
DEP. VAR. -4.9433D-02  1.4841D-01 -3.0404D-01  3.8588D 00
```

Figure 4-8r Monte Carlo Mode Check Problem Output

MONTE CARLO RESULTS   SAMPLE = 100   INTERNAL REJECTS = 12   USER REJECTS = 0

DEPENDENT VARIABLES WITH INTERNAL RANGES

CODE NUMBER = 306   MEAN = -0.239430D 01   VAR = 0.480705D 01   INTERVAL = 0.113723D 01
NO. REJECTS = 1   (MEAN) = -0.389337D 01   (VAR) = 0.143698D 02   TOLERANCE = 0.113723D 02

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.146970D 02	-0.152656D 02	-0.141284D 02	0.	0.0
-0.135598D 02	-0.141284D 02	-0.129912D 02	0.	0.0
-0.124226D 02	-0.129912D 02	-0.118539D 02	0.	0.0
-0.112853D 02	-0.118539D 02	-0.107167D 02	0.	0.0
-0.101481D 02	-0.107167D 02	-0.957950D 01	0.	0.0
-0.901098D 01	-0.957950D 01	-0.844227D 01	2.	0.199848D-01
-0.787366D 01	-0.844227D 01	-0.730504D 01	5.	0.499621D-01
-0.673643D 01	-0.730504D 01	-0.616782D 01	0.	0.0
-0.559921D 01	-0.616782D 01	-0.503059D 01	3.	0.299773D-01
-0.446198D 01	-0.503059D 01	-0.389337D 01	7.	0.699470D-01
-0.332476D 01	-0.389337D 01	-0.275614D 01	7.	0.699470D-01
-0.218753D 01	-0.275614D 01	-0.161892D 01	23.	0.229826D 00
-0.105030D 01	-0.161892D 01	-0.481692D 00	31.	0.309765D 00
0.683229D-01	-0.481692D 00	0.655534D 00	10.	0.999242D-01
0.122415D 01	0.655574D 00	0.179276D 01	0.	0.0
0.235137D 01	0.179276D 01	0.292998D 01	0.	0.0
0.348860D 01	0.292998D 01	0.406721D 01	0.	0.0
0.463582D 01	0.406721D 01	0.520444D 01	0.	0.0
0.577305D 01	0.520444D 01	0.634166D 01	0.	0.0
0.691027D 01	0.634166D 01	0.747889D 01	0.	0.0

CODE NUMBER = 307   MEAN = -0.188753D 00   VAR = 0.818949D 01   INTERVAL = 0.922790D 00
NO. REJECTS = 0   (MEAN) = 0.0   (VAR) = 0.946157D 01   TOLERANCE = 0.922790D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.876551D 01	-0.922790D 01	-0.830511D 01	0.	0.0
-0.784372D 01	-0.830511D 01	-0.738232D 01	0.	0.0
-0.692093D 01	-0.738232D 01	-0.645953D 01	0.	0.0
-0.599814D 01	-0.645953D 01	-0.553674D 01	2.	0.246289D-01
-0.507535D 01	-0.553674D 01	-0.461395D 01	2.	0.246289D-01
-0.415256D 01	-0.461395D 01	-0.369116D 01	6.	0.738866D-01
-0.322977D 01	-0.369116D 01	-0.276837D 01	6.	0.738866D-01
-0.230698D 01	-0.276837D 01	-0.184558D 01	9.	0.110830D 00
-0.138419D 01	-0.184558D 01	-0.922790D 00	11.	0.135459D 00
-0.461395D 00	-0.922790D 00	-0.693889D-16	15.	0.184716D 00
.461395D 00	-0.158207D-14	0.922790D 00	8.	0.985155D-01
0.138419D 01	0.922790D 00	0.184558D 01	8.	0.985155D-01
0.230698D 01	0.184558D 01	0.276837D 01	8.	0.985155D-01
0.322977D 01	0.276837D 01	0.369116D 01	6.	0.738856D-01
0.415256D 01	0.369116D 01	0.461395D 01	1.	0.123144D-01
0.507535D 01	0.461395D 01	0.553674D 01	3.	0.369433D-01
0.579814D 01	0.553674D 01	0.645953D 01	1.	0.123144D-01
0.672093D 01	0.645953D 01	0.738232D 01	1.	0.123144D-01
0.784372D 01	0.738232D 01	0.830511D 01	1.	0.123144D-01
0.876551D 01	0.830511D 01	0.922790D 01	0.	0.0

CODE NUMBER = 308   MEAN = -0.104863D-01   VAR = 0.820565D 01   INTERVAL = 0.922790D 00
NO. REJECTS = 0   (MEAN) = 0.167552D-16   (VAR) = 0.946157D 01   TOLERANCE = 0.922790D 01

Figure 4-8s Monte Carlo Mode Check Problem Output

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.876651D 01	-0.922790D 01	-0.830511D 01	0.	0.0
-0.784372D 01	-0.830511D 01	-0.738232D 01	0.	0.0
-0.692093D 01	-0.738232D 01	-0.645953D 01	2.	0.246289D-01
-0.599814D 01	-0.645953D 01	-0.553674D 01	1.	0.123144D-01
-0.507535D 01	-0.553674D 01	-0.461395D 01	0.	0.0
-0.415256D 01	-0.461395D 01	-0.369116D 01	5.	0.615722D-01
-0.322977D 01	-0.369116D 01	-0.276837D 01	7.	0.862010D-01
-0.230698D 01	-0.276837D 01	-0.184558D 01	8.	0.985155D-01
-0.138419D 01	-0.184558D 01	-0.922790D 00	8.	0.985155D-01
-0.461395D 00	-0.922790D 00	-0.416334D-16	18.	0.221650D 00
0.461395D 00	-0.156910D-14	0.922790D 00	6.	0.738866D-01
0.138419D 01	0.922790D 00	0.184558D 01	8.	0.985155D-01
0.230698D 01	0.184558D 01	0.276837D 01	10.	0.123144D 00
0.322977D 01	0.276837D 01	0.369116D 01	9.	0.110830D 00
0.415256D 01	0.369116D 01	0.461395D 01	2.	0.246289D-01
0.507535D 01	0.461395D 01	0.553674D 01	1.	0.123144D-01
0.599814D 01	0.553674D 01	0.645953D 01	1.	0.123144D-01
0.692093D 01	0.645953D 01	0.738232D 01	1.	0.123144D-01
0.784372D 01	0.738232D 01	0.830511D 01	1.	0.123144D-01
0.876651D 01	0.830511D 01	0.922790D 01	0.	0.0

CODE NUMBER = 320 MEAN = 0.452025D 01 VAR = 0.215039D 01 INTERVAL = 0.669580D 00
NO. REJECTS = 4 (MEAN) = 0.528874D 01 (VAR) = 0.498152D 01 TOLERANCE = 0.669580D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.107227D 01	-0.140706D 01	-0.737475D 00	0.	0.0
-0.402685D 00	-0.737475D 00	-0.678954D-01	0.	0.0
0.266894D 00	-0.678954D-01	0.601684D 00	0.	0.0
0.936474D 00	0.601684D 00	0.127126D 01	0.	0.0
0.150605D 01	0.127126D 01	0.194084D 01	0.	0.0
0.227563D 01	0.194084D 01	0.261042D 01	0.	0.0
0.294521D 01	0.261042D 01	0.328000D 01	0.	0.0
0.361479D 01	0.328000D 01	0.394958D 01	43.	0.729766D 00
0.428437D 01	0.394958D 01	0.461916D 01	23.	0.390340D 00
0.495395D 01	0.461916D 01	0.528874D 01	7.	0.118799D 00
0.562353D 01	0.528874D 01	0.595832D 01	5.	0.848565D-01
0.629311D 01	0.595832D 01	0.662790D 01	3.	0.509139D-01
0.696269D 01	0.662790D 01	0.729748D 01	0.	0.0
0.763227D 01	0.729748D 01	0.796706D 01	0.	0.0
0.830185D 01	0.796706D 01	0.863664D 01	3.	0.509139D-01
0.897143D 01	0.863664D 01	0.930622D 01	3.	0.509139D-01
0.964101D 01	0.930622D 01	0.997580D 01	0.	0.0
0.103106D 02	0.997580D 01	0.106454D 02	1.	0.169713D-01
0.109802D 02	0.106454D 02	0.113150D 02	0.	0.0
0.116498D 02	0.113150D 02	0.119845D 02	0.	0.0

CODE NUMBER = 317 MEAN = 0.230931D-03 VAR = 0.459456D-07 INTERVAL = 0.109095D-03
NO. REJECTS = 0 (MEAN) = 0.373499D-03 (VAR) = 0.132242D-06 TOLERANCE = 0.109095D-02

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.662905D-03	-0.717453D-03	-0.608358D-03	0.	0.0
-0.553810D-03	-0.608358D-03	-0.499263D-03	0.	0.0
-0.444715D-03	-0.499263D-03	-0.390167D-03	0.	0.0
-0.335620D-03	-0.390167D-03	-0.281072D-03	0.	0.0
-0.226525D-03	-0.281072D-03	-0.171977D-03	0.	0.0
-0.117430D-03	-0.171977D-03	-0.628820D-04	0.	0.0
-0.833442D-05	-0.628820D-04	0.462132D-04	10.	0.104163D 04

Figure 4-8t Monte Carlo Mode Check Problem Output

0.130761D-03	0.462132D-04	[illegible]155309D-03	31.	0.322904D 04
0.229855D-03	0.155308D-03	[illegible]264403D-03	23.	0.239574D 04
0.318951D-03	0.264403D-03	[illegible]73499D-03	7.	0.729138D [illegible]
0.428046D-03	0.373499D-03	[illegible]25940-03	7.	0.729138[illegible]
0.537141D-03	0.482594D-03	0 [illegible]1659D-03	3.	0.312488D
0.646236D-03	0.591689D-03	0.[illegible]007840-03	0.	0.0
0.755332D-03	0.700784D-03	0.[illegible]09879D-03	5.	0.520813D 03
0.864427D-03	0.809879D-03	0.919974D-03	1.	0.104163D 03
0.973522D-03	0.918974D-03	[illegible]102807D-02	1.	0.104163D 03
0.108262D-02	0.102807D-02	[illegible]37160-02	0.	0.0
0.119171D-02	0.113716D-02	[illegible]6260-02	0.	0.0
0.130081D-02	[illegible]46260-02	[illegible]5350-02	0.	0.0
0.140990D-02	[illegible]535D-02	0 [illegible]445D-02	0.	0.0

CODE NUMBER = 322 MEAN = 0.141799D 01 VAR = 0.479753D 01 INTERVAL = 0.113723D 01
NO. REJECTS = 0 (MEAN) = -0.389337D 01 (VAR) = 0.143698D 02 TOLERANCE = 0.113723D 02

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.145270D 02	-0.152656D 02	-0.141284D 02	0.	0.0
-0.135598D 02	-0.141284D 02	-0.129912D 02	0.	0.0
-0.124226D 02	-0.129912D 02	-0.118539D 02	0.	0.0
-0.112857D 02	-0.118539D 02	-0.107167D 02	0.	0.0
-0.101481D 02	-0.107167D 02	-0.957950D 01	0.	0.0
-0.901088D 01	-0.957950D 01	-0.844227D 01	0.	0.0
-0.787365D 01	-0.844227D 01	-0.730504D 01	0.	0.0
-0.673643D 01	-0.730504D 01	-0.616782D 01	0.	0.0
-0.559921D 01	-0.616782D 01	-0.503059D 01	1.	0.999242D-02
-0.446198D 01	-0.503059D 01	-0.389337D 01	5.	0.499621D-01
-0.332476D 01	-0.389337D 01	-0.275614D 01	1.	0.999242D-02
-0.218753D 01	-0.275614D 01	-0.161892D 01	1.	0.999242D-02
-0.105030D 01	-0.161892D 01	-0.481692D 00	5.	0.499621D-01
0.859209D-01	-0.481692D 00	0.655534D 00	8.	0.799394D-01
0.122415D 01	0.655534D 00	0.179276D 01	18.	0.179864D 00
0.236137D 01	0.179276D 01	0.292998D 01	27.	0.269795D 00
0.349860D 01	0.292998D 01	0.406721D 01	22.	0.219833D 00
0.463582D 01	0.406721D 01	0.520444D 01	0.	0.0
0.577305D 01	0.520444D 01	0.634166D 01	0.	0.0
0.691027D 01	0.634166D 01	0.747889D 01	0.	0.0

CODE NUMBER = 323 MEAN = -0.188753D 00 VAR = 0.818949D 01 INTERVAL = 0.922790D 00
NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.946157D 01 TOLERANCE = 0.922790D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.876651D 01	-0.922790D 01	-0.830511D 01	0.	0.0
-0.784372D 01	-0.830511D 01	-0.738232D 01	0.	0.0
-0.692093D 01	-0.738232D 01	-0.645953D 01	0.	0.0
-0.599814D 01	-0.645953D 01	-0.553674D 01	2.	0.246289D-01
-0.507535D 01	-0.553674D 01	-0.461395D 01	2.	0.246289D-01
-0.415256D 01	-0.461395D 01	-0.369116D 01	6.	0.738866D-01
-0.322977D 01	-0.369116D 01	-0.276837D 01	6.	0.738866D-01
-0.230698D 01	-0.276837D 01	-0.184558D 01	9.	0.110830D 00
-0.138419D 01	-0.184558D 01	-0.922790D 00	11.	0.135459D 00
-0.461395D 00	-0.922790D 00	-0.693889D-16	15.	0.184716D 00
0.461395D 00	-0.159207D-14	0.922790D 00	8.	0.985155D-01
0.138419D 01	0.922790D 00	0.184558D 01	8.	0.985155D-01
0.230698D 01	0.184558D 01	0.276837D 01	8.	0.985155D-01
0.322977D 01	0.276837D 01	0.369116D 01	6.	0.738866D-01
0.415256D 01	0.369116D 01	0.461395D 01	1.	0.123144D-01
0.507535D 01	0.461395D 01	0.553674D 01	3.	0.369433D-01

Figure 4-8u Monte Carlo Mode Check Problem Output

```
0.599814D 01  0.553674D 01  0.645953D 01   1.  0.123144D-01
0.692093D 01  0.645953D 01  0.738232D 01   1.  0.123144D-01
0.784372D 01  0.738232D 01  0.830511D 01   1.  0.123144D-01
0.876651D 01  0.830511D 01  0.922790D 01   0.  0.0

CODE NUMBER =  324    MEAN = -0.104863D-01    VAR = 0.820565D 01    INTERVAL = 0.922790D 00
NO. REJECTS =    0  (MEAN) =  0.167552D-16  (VAR) = 0.946157D 01   TOLERANCE = 0.922790D 01

  CLASS MARK    LOWER BOUND   UPPER BOUND FREQ   DIST. FUNCT.
-0.876651D 01 -0.922790D 01 -0.830511D 01   0.  0.0
-0.784372D 01 -0.830511D 01 -0.738232D 01   0.  0.0
-0.692093D 01 -0.738232D 01 -0.645953D 01   2.  0.246288D-01
-0.599814D 01 -0.645953D 01 -0.553674D 01   1.  0.123144D-01
-0.507535D 01 -0.553674D 01 -0.461395D 01   0.  0.0
-0.415256D 01 -0.461395D 01 -0.369116D 01   5.  0.615722D-01
-0.322977D 01 -0.369116D 01 -0.276837D 01   7.  0.862010D-01
-0.230698D 01 -0.276837D 01 -0.184558D 01   8.  0.985155D-01
-0.138419D 01 -0.184558D 01 -0.922790D 00   8.  0.985155D-01
-0.461395D 00 -0.922790D 00 -0.416334D-16  18.  0.221660D 00
 0.461395D 00 -0.156819D-14  0.922790D 00   6.  0.738866D-01
 0.138419D 01  0.922790D 00  0.184558D 01   8.  0.995155D-01
 0.230698D 01  0.184558D 01  0.276837D 01  10.  0.123144D 00
 0.322977D 01  0.276837D 01  0.369116D 01   9.  0.110830D 00
 0.415256D 01  0.369116D 01  0.461395D 01   2.  0.246288D-01
 0.507535D 01  0.461395D 01  0.553674D 01   1.  0.123144D-01
 0.599814D 01  0.553674D 01  0.645953D 01   1.  0.123144D-01
 0.692093D 01  0.645953D 01  0.738232D 01   1.  0.123144D-01
 0.784372D 01  0.738232D 01  0.830511D 01   1.  0.123144D-01
 0.876651D 01  0.830511D 01  0.922790D 01   0.  0.0

CODE NUMBER =  518    MEAN = 0.109761D 00    VAR = 0.459456D-07    INTERVAL = 0.109095D-03
NO. REJECTS =    0  (MEAN) = 0.109904D 00  (VAR) = 0.132242D-06   TOLERANCE = 0.109095D-02

  CLASS MARK    LOWER BOUND   UPPER BOUND FREQ   DIST. FUNCT.
 0.108867D 00  0.108813D 00  0.108922D 00   0.  0.0
 0.108977D 00  0.108922D 00  0.109031D 00   0.  0.0
 0.109086D 00  0.109031D 00  0.109140D 00   0.  0.0
 0.109195D 00  0.109140D 00  0.109249D 00   0.  0.0
 0.109304D 00  0.109249D 00  0.109358D 00   0.  0.0
 0.109413D 00  0.109359D 00  0.109467D 00   0.  0.0
 0.109522D 00  0.109467D 00  0.109577D 00  10.  0.104163D 04
 0.109631D 00  0.109577D 00  0.109686D 00  31.  0.322904D 04
 0.109740D 00  0.109686D 00  0.109795D 00  23.  0.239574D 04
 0.109849D 00  0.109795D 00  0.109904D 00   7.  0.729138D 03
 0.109958D 00  0.109904D 00  0.110013D 00   7.  0.729138D 03
 0.110067D 00  0.110013D 00  0.110122D 00   3.  0.312488D 03
 0.110177D 00  0.110122D 00  0.110231D 00   0.  0.0
 0.110286D 00  0.110231D 00  0.110340D 00   5.  0.520813D 03
 0.110395D 00  0.110340D 00  0.110449D 00   1.  0.104163D 03
 0.110504D 00  0.110449D 00  0.110558D 00   1.  0.104163D 03
 0.110613D 00  0.110558D 00  0.110668D 00   0.  0.0
 0.110722D 00  0.110668D 00  0.110777D 00   0.  0.0
 0.110831D 00  0.110777D 00  0.110886D 00   0.  0.0
 0.110940D 00  0.110886D 00  0.110995D 00   0.  0.0

CODE NUMBER =  510    MEAN = 0.103940D 05    VAR = 0.151009D 04    INTERVAL = 0.129803D 02
NO. REJECTS =    7  (MEAN) = 0.103815D 05  (VAR) = 0.157208D 04   TOLERANCE = 0.129803D 03
```

Figure 4-8v Monte Carlo Mode Check Problem Output

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0.102582D 05	0.102517D 05	0.102647D 05	1.	0.875456D-03
0.102712D 05	0.102647D 05	0.102777D 05	1.	0.875456D-03
0.102842D 05	0.102777D 05	0.102907D 05	4.	0.350182D-02
0.102972D 05	0.102907D 05	0.103037D 05	1.	0.875456D-03
0.103101D 05	0.103037D 05	0.103166D 05	0.	0.0
0.103231D 05	0.103166D 05	0.103296D 05	2.	0.175091D-02
0.103361D 05	0.103296D 05	0.103426D 05	1.	0.875456D-03
0.103491D 05	0.103426D 05	0.103556D 05	5.	0.437728D-02
0.103621D 05	0.103556D 05	0.103686D 05	6.	0.525274D-02
0.103750D 05	0.103686D 05	0.103815D 05	9.	0.787910D-02
0.103880D 05	0.103815D 05	0.103945D 05	12.	0.105055D-01
0.104010D 05	0.103945D 05	0.104075D 05	19.	0.166337D-01
0.104140D 05	0.104075D 05	0.104205D 05	19.	0.166337D-01
0.104270D 05	0.104205D 05	0.104335D 05	8.	0.700365D-02
0.104400D 05	0.104335D 05	0.104464D 05	0.	0.0
0.104529D 05	0.104464D 05	0.104594D 05	0.	0.0
0.104659D 05	0.104594D 05	0.104724D 05	0.	0.0
0.104789D 05	0.104724D 05	0.104854D 05	0.	0.0
0.104919D 05	0.104854D 05	0.104984D 05	0.	0.0
0.105049D 05	0.104984D 05	0.105113D 05	0.	0.0

INDEPENDENT VARIABLES WITH INTERNAL RANGES

CODE NUMBER = 116 MEAN = -0.102273D-01 VAR = 0.156671D 01 INTERVAL = 0.300000D 00
NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.100000D 01 TOLERANCE = 0.300000D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.285000D 01	-0.300000D 01	-0.270000D 01	3.	0.113636D 00
-0.255000D 01	-0.270000D 01	-0.240000D 01	1.	0.378788D-01
-0.225000D 01	-0.240000D 01	-0.210000D 01	2.	0.757576D-01
-0.195000D 01	-0.210000D 01	-0.180000D 01	1.	0.378788D-01
-0.165000D 01	-0.180000D 01	-0.150000D 01	3.	0.113636D 00
-0.135000D 01	-0.150000D 01	-0.120000D 01	3.	0.113636D 00
-0.105000D 01	-0.120000D 01	-0.900000D 00	5.	0.189394D 00
-0.750000D 00	-0.900000D 00	-0.600000D 00	6.	0.227273D 00
-0.450000D 00	-0.600000D 00	-0.300000D 00	5.	0.189394D 00
-0.150000D 00	-0.300000D 00	-0.138778D-16	19.	0.719697D 00
0.150000D 00	-0.527356D-15	0.300000D 00	12.	0.454545D 00
0.450000D 00	0.300000D 00	0.600000D 00	3.	0.113636D 00
0.750000D 00	0.600000D 00	0.900000D 00	9.	0.340909D 00
0.105000D 01	0.900000D 00	0.120000D 01	2.	0.757576D-01
0.135000D 01	0.120000D 01	0.150000D 01	1.	0.378788D-01
0.165000D 01	0.150000D 01	0.180000D 01	5.	0.189394D 00
0.195000D 01	0.180000D 01	0.210000D 01	2.	0.757576D-01
0.225000D 01	0.210000D 01	0.240000D 01	3.	0.113636D 00
0.255000D 01	0.240000D 01	0.270000D 01	1.	0.378788D-01
0.285000D 01	0.270000D 01	0.300000D 01	2.	0.757576D-01

CODE NUMBER = 117 MEAN = -0.255682D 00 VAR = 0.139304D 01 INTERVAL = 0.300000D 00
NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.100000D 01 TOLERANCE = 0.300000D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.285000D 01	-0.300000D 01	-0.270000D 01	5.	0.189394D 00
-0.255000D 01	-0.270000D 01	-0.240000D 01	0.	0.0
-0.225000D 01	-0.240000D 01	-0.210000D 01	1.	0.378788D-01
-0.195000D 01	-0.210000D 01	-0.180000D 01	1.	0.378788D-01

Figure 4-8w Monte Carlo Mode Check Problem Output

-0.165000D 01	-0.180000D 01	-0.150000D 01	3.	0.113636D 00
-0.135000D 01	-0.150000D 01	-0.120000D 01	4.	0.151515D 00
-0.105000D 01	-0.120000D 01	-0.900000D 00	10.	0.378788D 00
-0.750000D 00	-0.900000D 00	-0.600000D 00	9.	0.340909D 00
-0.450000D 00	-0.600000D 00	-0.300000D 00	9.	0.340909D 00
-0.150000D 00	-0.300000D 00	-0.138778D-16	10.	0.378788D 00
0.150000D 00	-0.527356D-15	0.300000D 00	12.	0.454545D 00
0.450000D 00	0.300000D 00	0.600000D 00	7.	0.265152D 00
0.750000D 00	0.600000D 00	0.900000D 00	6.	0.227273D 00
0.105000D 01	0.900000D 00	0.120000D 01	4.	0.151515D 00
0.135000D 01	0.120000D 01	0.150000D 01	1.	0.378788D-01
0.165000D 01	0.150000D 01	0.180000D 01	0.	0.0
0.195000D 01	0.180000D 01	0.210000D 01	2.	0.757576D-01
0.225000D 01	0.210000D 01	0.240000D 01	2.	0.757576D-01
0.255000D 01	0.240000D 01	0.270000D 01	0.	0.0
0.285000D 01	0.270000D 01	0.300000D 01	2.	0.757576D-01

CODE NUMBER = 118 MEAN = -0.582955D 01 VAR = 0.725602D 04 INTERVAL = 0.270000D 02
NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.810000D 04 TOLERANCE = 0.270000D 03

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.256500D 03	-0.270000D 03	-0.243000D 03	0.	0.0
-0.229500D 03	-0.243000D 03	-0.216000D 03	0.	0.0
-0.202500D 03	-0.216000D 03	-0.189000D 03	1.	0.420875D-03
-0.175500D 03	-0.189000D 03	-0.162000D 03	2.	0.841751D-03
-0.148500D 03	-0.162000D 03	-0.135000D 03	1.	0.420875D-03
-0.121500D 03	-0.135000D 03	-0.108000D 03	6.	0.252525D-02
-0.945000D 02	-0.108000D 03	-0.810000D 02	6.	0.252525D-02
-0.675000D 02	-0.810000D 02	-0.540000D 02	9.	0.378788D-02
-0.405000D 02	-0.540000D 02	-0.270000D 02	11.	0.462963D-02
-0.135000D 02	-0.270000D 02	0.0	15.	0.631313D-02
0.135000D 02	-0.479616D-13	0.270000D 02	8.	0.336700D-02
0.405000D 02	0.270000D 02	0.540000D 02	7.	0.294613D-02
0.675000D 02	0.540000D 02	0.810000D 02	9.	0.378788D-02
0.945000D 02	0.810000D 02	0.108000D 03	6.	0.252525D-02
0.121500D 03	0.108000D 03	0.135000D 03	1.	0.420875D-03
0.148500D 03	0.135000D 03	0.162000D 03	3.	0.126263D-02
0.175500D 03	0.162000D 03	0.189000D 03	1.	0.420875D-03
0.202500D 03	0.189000D 03	0.216000D 03	1.	0.420875D-03
0.229500D 03	0.216000D 03	0.243000D 03	1.	0.420875D-03
0.256500D 03	0.243000D 03	0.270000D 03	0.	0.0

CODE NUMBER = 119 MEAN = -0.539568D-13 VAR = 0.714089D 04 INTERVAL = 0.270000D 02
NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.810000D 04 TOLERANCE = 0.270000D 03

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
-0.256500D 03	-0.270000D 03	-0.243000D 03	0.	0.0
-0.229500D 03	-0.243000D 03	-0.216000D 03	0.	0.0
-0.202500D 03	-0.216000D 03	-0.189000D 03	2.	0.841751D-03
-0.175500D 03	-0.189000D 03	-0.162000D 03	1.	0.420875D-03
-0.148500D 03	-0.162000D 03	-0.135000D 03	0.	0.0
-0.121500D 03	-0.135000D 03	-0.108000D 03	6.	0.252525D-02
-0.945000D 02	-0.108000D 03	-0.810000D 02	6.	0.252525D-02
-0.675000D 02	-0.810000D 02	-0.540000D 02	8.	0.336700D-02
-0.405000D 02	-0.540000D 02	-0.270000D 02	8.	0.336700D-02
-0.135000D 02	-0.270000D 02	0.0	18.	0.757576D-02
0.135000D 02	-0.479616D-13	0.270000D 02	5.	0.210438D-02
0.405000D 02	0.270000D 02	0.540000D 02	8.	0.336700D-02
0.675000D 02	0.540000D 02	0.810000D 02	11.	0.462963D-02

Figure 4-8x Monte Carlo Mode Check Problem Output

0.9450000 02	0.8100000 02	0.1080000 03	9.	0.3787680-02
0.1215000 03	0.1080000 03	0.1350000 03	2.	0.8417510-03
0.1485000 03	0.1350000 03	0.1620000 03	1.	0.4208750-03
0.1755000 03	0.1620000 03	0.1890000 03	1.	0.4208750-03
0.2025000 03	0.1890000 03	0.2160000 03	1.	0.4208750-03
0.2295000 03	0.2160000 03	0.2430000 03	1.	0.4208750-03
0.2565000 03	0.2430000 03	0.2700000 03	0.	0.0

Figure 4-8y. Monte Carlo Mode Check Problem Output

Table 4-1 Monte Carlo Mode Output Interpretation

VARIABLE NAME	DESCRIPTION	COMMENTS
CØDE Number	Independent/dependent variable identification number.	• See Table 3-3 for definitions. • CØDE NUMBER list is identical to CØDE# list defined by Card Group 2.
MEAN	Expected or mean value of the variable as determined by histogram.	• See Eq. (B.1-22) for algorithm.
VA[illegible]	Variance of the variable as determined by the histogram.	• See Eq. (B.1-24) for algorithm.
INTERVAL	Histogram cell size: $\text{INTERVAL} = \frac{2 * \text{TØLERANCE}}{\text{NCELL}}$	• NCELL is the number of cells as defined by Card Group 1. • TØLERANCE is defined later in this table.
NØ. REJECTS	Number of Monte Carlo experiments rejected from all histograms due to range limitations on this variable.	• Rejects occur only on dependent variables (i.e., TYPE = 7 or TYPE = 8) • See Appendix B.2.3, paragraph Monte Carlo, for a discussion of rejects.
(MEAN)	Defines midpoint of histogram. The value is generally related to the mean or expected value of the variable. However, it is subject to varying definitions.	
	TYPE = 2, 3 — Mean or expected value of variable as determined by VALUE on Group 2 card.	• For TYPE = 2 or TYPE = 3, the input mean value is repeated for reference purposes.
	TYPE = 4 — Midpoint of abscissa table used to define the probability density function.	• For TYPE = 4, the actual mean value may be determined from the LIMITING CASES printed output. VALUE (as given there) is the mean value determined by the probability density function.

Table 4-1 Monte Carlo Mode Output Interpretation (cont'd)

VARIABLE NAME	DESCRIPTION	COMMENTS
(MEAN) (cont'd)	TYPE = 7 The expected or mean value as determined by the Statistical Processor.	• See Eq. (B.1-5) for algorithm. • Details of the calculation depend on the IØPRNT variable defined by Group 1 Card.
	TYPE = 8 Definition depends on histogram range: -<u>INTERNAL RANGES</u> - TYPE = 7 definition applies. -<u>USER RANGES</u> - Equal to VALUE as defined by Group 2 Card.	
(VAR)	Value sometimes related to variance (σ^2) of the variable. However, it is subject to varying definitions.	
	TYPE = 2, 3 Variance of variable as determined by STDEV on Group 2 Card: (VAR) = STDEV**2	• For TYPE = 2 or TYPE = 3, the input variance is repeated for reference purposes.
	TYPE = 4 Maximum tabulated value of probability density function.	• For TYPE = 4, the actual variance (σ^2) may be determined from the LIMITING CASES printed output. TØLERANCE (as defined there) is related to the standard deviation (σ) determined by the input probability density function according to σ = 3 * TØLERANCE so that the variance is σ^2 = 9 * TØLERANCE**2 • Maximum value determined by search of probability density function table and used in generating random samples. See Appendix A.5.3 for full discussion.

Table 4-1 Monte Carlo Mode Output Interpretation (cont'd)

VARIABLE NAME	DESCRIPTION		COMMENTS
(VAR) (cont'd)	TYPE = 7	The variance (σ^2) as determined by the Statistical Processor.	• See Eq. (B.1-7) for algorithm. • Details of calculation depend on IØPRNT variable defined by Group 1 card.
	TYPE = 8	Definition depends on histogram range: -<u>INTERNAL RANGES</u> - TYPE = 7 definition applies. -USER RANGES - VALUE determined by STDEV value on Group 2 Card according to (VAR) = STDEV**2	
TØLERANCE	Defines the histogram half range. Subject to varying definitions.		
	TYPE = 2	The larger of 3*STDEV or TØL, both as defined on the Group 2 Card.	• Normally TØL is selected to be small for the purposes of evaluating the derivatives. Thus, the half range of the histogram will usually be TØLERANCE = 3*STDEV
	TYPE = 3	The value of TØL as defined by the Group 2 Card.	
	TYPE = 4	Half the interval covered by the probability density function abscissa table.	

Table 4-1 Monte Carlo Mode Output Interpretation (concl'd)

VARIABLE NAME	DESCRIPTION	COMMENTS
TØLERANCE (cont'd)	TYPE = 7 — The 3σ value where σ is defined by the Statistical Processor, i.e., $\sigma = \sqrt{(VAR)}$ TYPE = 8 — Definition depends on histogram range: -INTERNAL RANGES - TYPE = 7 definition applies. -USER RANGES - TØL as defined by the Group 2 Card.	
CLASS MARK	Midpoint of histogram cells.	
LØWER BØUND	Left hand cell boundary.	
UPPER BØUND	Right hand cell boundary.	
FREQ.	Number of occurrences of cell during Monte Carlo experiments.	• Total of FREQ Column is equal to the number of trials (NTRIAL on Group 1 Card) less the number of rejects.
DIST.FUNCT.	Estimated probability density function value for this interval.	• See Eq. (B.1-18) for computational algorithm.

5.0 SUMMARY

This report is a User's Manual for the Hypervelocity[1] Inflight Trajectory Scatter (HITS) computer code. The primary purpose of the code is to evaluate "projectile dispersion" which is defined here as the miss distance associated with inflight behavior not anticipated by Fire Control. Projectile dispersion poses a fundamental limit on overall weapon system effectiveness. The code is central to a scientific, systematic approach to evaluating projectile dispersion. The method consists of (1) constructing a comprehensive model for each source of projectile dispersion (i.e., the error source model) and (2) processing it through the HITS code to obtain a detailed projectile dispersion error budget, replete with sensitivity coefficients. The error budget identifies the larger dispersion sources and those with significant potential, as well as the total dispersion. Both crossrange and downrange dispersion are computed. The error budget defines dispersion on a level conducive to interpretation and a comprehensive understanding.

The HITS code evaluates projectile dispersion statistics by manipulating the inputs to a set of trajectory equations representing the true trajectory and comparing the computed flight path to the trajectory predicted by a second set of equations representing Fire Control. The true trajectory is varied in accord with the error source model; the Fire Control Trajectory is based on nominal projectile characteristics. At the selection of the user, variations are governed by either analytical or Monte Carlo statistical methods.

The two sets of trajectory equations are closed form approximations to the full six degree of freedon (6 DOF) equations of motion. They consist of (1) a particle trajectory with a velocity dependent drag coefficient and (2) two perturbation equations which correct for angle of attack effects. Extensive 6 DOF simulations have been conducted to verify the approximations. These equations also have potential application tc operational Fire Control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

[1] "Hypervelocity" is used throughout this report to indicate the limitation of the present HITS code to projectiles whose velocity does not become transonic at any point along the trajectory. No other connotation is intended.

The HITS code consists of two portions. One portion performs all input/output functions and the statistical calculations. This portion is an adaptation of existing AVCO software (i.e., YP-58). It brings to the HITS code additional capability as a projectile design tool; a capability the code would not otherwise have had. The closed form trajectory equations constitute the second portion of the code. These equations were specifically developed for dispersion assessment under an earlier contract.[1] The code was assembled under the earlier contract and used to evaluate the projectile dispersion of a reference design. The code, however, was developed only to the level of a research program and documented only to the extent necessary to support the analyses of the study. Detailed input/output information was not presented.

Under the present effort, the HITS code was advanced to the level of a production code. The code is maintained in the AVCO engineering computer code library as Production Code 5127. This report is the associated User's Manual. It places maximum emphasis on input/output information. The text contains all the day-to-day necessities required to operate the code and interpret the results of computation. Input/output information is summarized by tables featuring a quick-reference step-by-step format. Example problems are presented and discussed. They verify proper installation and illustrate the output format. The appendices discuss theoretical and programming aspects of the code. They contain the theoretical development of the closed form trajectory equations and a complete listing of the code.

---

[1]Gustafson, T. G., Crimi, P., Bellaire, R. G., "Dispersion of Increased Velocity Projectiles - Feasibility Phase," Avco Corporation, AVSD-0200-75-CR, July 1975.

APPENDIX A

ELEMENTS OF PROBABILITY THEORY AND STATISTICS

This appendix presents an overview of the pertinent aspects of probability theory. Sections A.1 through A.5 review basic concepts and theoretical relationships. Sections A.5 and A.6 relate the theory to the practical problems associated with Monte Carlo methods. More detailed treatments may be found in text books.[1,2]

A.1 Random Phenomenon and Probabilities

Random phenomenon are those phenomena whose fundamental processes are either not completely understood or are too complex to define in an entirely satisfactory manner. In either case, the scientific interest in the phenomenon centers around the "average" rather than the "precise" outcome. Probability theory is a mathematical discipline for quantifying random phenomenon.

A.1.1 Fundamental Concepts

Probability theory has three building blocks: (1) the sample description space, (2) a collection of events, and (3) a probability function. This section discusses these concepts.

Sample Description Space

The sample description space is the collection of all possible outcomes of the random phenomena. Each occurrence is thought of as the result of an experiment. For instance, the sample description space for a coin flipping experiment would consist of the two possible outcomes (H, T). The sample description space for a temperature measurement taken at randomly selected time and location would include all values from $-\infty$ to $+\infty$, $(-\infty, +\infty)$.

Events

Random events (or simply "Events") are the groupings of the fundamental outcomes. They are subsets of the sample description space. Events are defined by the analyst. For instance, a meteorologist interested in the likelihood of sub-freezing temperatures would

---

[1] Parzen, E., Modern Probability Theory and its Applications, Wiley, New York, 1960.

[2] Davenport, W. B., and Root, W. L., An Introduction to the Theory of Random Signals and Noise, McGraw-Hill, New York, 1958.

select only two events to subdivide the sample description space of all possible temperatures: $(-\infty, 32)$ and $(32, +\infty)$.

Probability Function

The probability function, P, assigns to each event a number between zero and one corresponding to the relative likelihood of occurrence of the event. The probability of an event, A, is the [illegible]etical ratio of the number of times it would occur, N, to the [illegible] number of experiments, N_T.

$$P[A] = \frac{N}{N_T} \tag{A.1-1}$$

(Monte Carlo procedures estimate probabilities by counting the results of experiments.) An event which (almost) never occurs has a probability of zero. An event which (almost) always occurs has a probability of one. For instance, if freezing temperatures occurred 40% of the time the probability of that event would be

$$P[(-\infty, 32)] = 0.40 \tag{A.1-2}$$

Venn Diagrams

A Venn diagram is a graphical method of depicting the relationship between the sample description space and events. (It also gives a convenient method for illustrating the rules of combination for probabilities.) Figure A-1 shows a Venn diagram. There are three events A, B, and C. Events A and B overlap so A and B simultaneously occur for some experimental outcomes. Event C is exclusive of A and B, since neither A nor B can occur when C occurs.

A.1.2 Rules of Combination

Let the probabilities of the individual events in Figure A-1 be P(A), P(B), and P(C). There are rules of combination. Let A + B denote the occurrence of either event, and let AB denote the simultaneous occurrence of both events. The probability of A + B is

$$P(A+B) = P(A) + P(B) - P(AB) \tag{A.1-3}$$

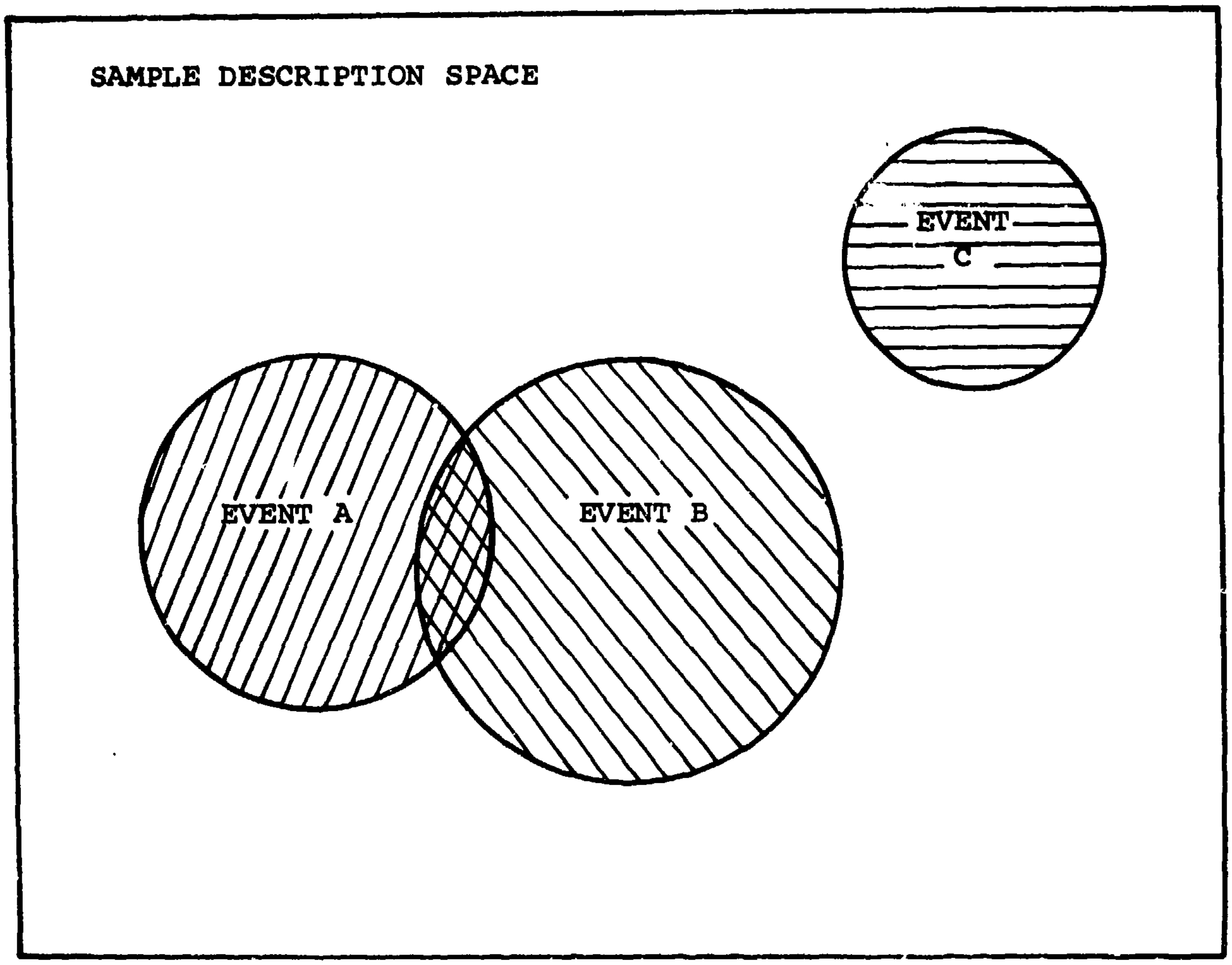


FIGURE A-1 THE VENN DIAGRAM

Referring to Figure A-1, A + B is the area encompassed by either events A or B, and AB is the area common to both A and B. With these visualizations, Eq. (A.1-3) simply states that P(A+B) is equal to P(A) + P(B) less a correction, P(AB), for the common area which is counted twice in the sum P(A) + P(B).

Since probabilities are positive numbers, Eq. (A.1-3) implies

$$P(A + B) \leq P(A) + P(B) \tag{A.1-4}$$

Considering a collection of events: A, B, C, ... with the probabilities P(A), P(B), P(C)...., the probability that at least one of the events will occur is P(A + B + C + ...) which is related to the individual probabilities through the inequality:

$$P(A + B + C + \ldots) \leq P(A) + P(B) + P(C) + \cdots \tag{A.1-5}$$

It is an inequality since it may be possible for at least two of the events to occur concurrently. If only one of the events can occur the events are said to be mutually exclusive and equality holds. If at least one of the events must occur, the ensemble of events is termed exhaustive in which case the left side of the above inequality is identically one.

A.1.3 Conditional Probabilities

Considering two events, A and B, the probability of their simultaneous or joint occurrence is P(AB). The probability P(A/B) is the conditional probability of A given B and is defined by:

$$P(A/B) = \frac{P(AB)}{P(B)} \tag{A.1-6}$$

provided P(B) > 0. The conditional probability P(A/B), is the probability A will occur given that B has occurred. Two events, A and B, are said to be statistically independent if knowledge that B has occurred is of no value in inferring that A will occur so P(A/B) = P(A). This means

$$P(ABC\ldots) = P(A)\,P(B)\,P(C)\ldots \tag{A.1-7}$$

if A, B, C, ... are mutually independent events. That is, for statistically independent events, the probability of simultaneous occurrence is equal to the product of the individual probabilities.

A.2 Random Variables

The discussion of the previous section dealt with random phenomena in general. The discussion from this point on is restricted to random variables, i.e., those whose outcomes are numerically valued. The outside temperature, the instantaneous voltage across a resistor, and the displacement of the impact point from the target point are just three examples of random variables.

A.2.1 Distribution Functions

A numerically valued random phenomenon is one which is completely summarized by the value of a descriptive variable (i.e., temperature, voltage, or misdistance in the foregoing examples). The variable is referred to as a random or stochastic variable. The sample description space is all real numbers, and the events are all intervals and combinations of intervals. The probability function is defined by the (probability) distribution function, F,

$$F(X) = P(x \leq X) \tag{A.2-1}$$

The distribution function evaluated at X is the probability that the random variable x has a value less than or at the most equal to X. By definition, the cumulative distribution function has the properties $F(-\infty) = 0$, $F(+\infty) = 1$ and $F(x)$ is a positive, nondecreasing function of x. (Note: It is customary to notationally distinguish between a random variable, x, and a value it could assume, X, only when demanded by clarity.) A distribution function need not be a continuous function. It may have step-type discontinuities at discrete points corresponding to values with a positive probability of occurrence. Random variables may be "discrete," "continuous," or "mixed." The remainder of this section discusses these types.

A discrete random variable can take on specific discrete values $X_1, X_2, \ldots X_n$. The numerical value shown on a pair of dice is an example of a discrete random variable. Each experiment must result in an integer value between two and twelve. If f_i is the probability, X_i occurs, the distribution function is given as

$$F(X) = \sum_{X_i \leq X} f_i = P(x \leq X) \tag{A.2-2}$$

and

$$F(X_N) = \sum_{i=1}^{N} f_i = 1 \tag{A.2-3}$$

If the distribution function derivative exists, the random variable is said to be continuous and the (probability) density function is defined according to

$$f(x) = \frac{d}{dx} F(x) \tag{A.2-4}$$

The density function at X, f(X), is the probability of the event that the random variable x falls in the range

$$X \leq x \leq X + dX \tag{A.2-5}$$

The probability of occurrence of any specific value of a continuous random variable is zero, since

$$\lim_{\epsilon \to 0} P[X \leq x \leq X + \epsilon] = \lim_{\epsilon \to 0} \int_{X}^{X+\epsilon} f(x)\, dx \tag{A.2-6}$$

Air temperature is an example of a continuous random variable since the likelihood of any specific temperature is zero. As a result of the definitions

$$\int_{-\infty}^{\infty} f(x)\, dx = F(\infty) - F(-\infty) = 1 \tag{A.2-7}$$

The distribution of a mixed random variable contains both discrete values, with positive probability of occurrence, and a continuous distribution.

Most random variables treated by Monte Carlo methods are continuous. The density function is estimated by dividing up the range of the random variable into small intervals called cells. Histograms are then constructed which count the number of times the value of the random variable falls in each cell, N_i. The exact value is discarded and only the histogram is maintained during execution of the code. The density function is estimated as a piece-wise constant function by the relationship

$$f(x) = \frac{N_i}{N_T} \frac{1}{x_{b_{i+1}} - x_{b_i}} \qquad \text{(A.2-8)}$$

where N_T is the total number of trials and $x_{b_{i+i}}$ and x_b are the upper and lower boundaries of the i^{th} cell. It might be argued that Monte Carlo methods, in effect, approximate all random variables as discrete since only the histogram data is saved. Nevertheless, it is most convenient to treat only continuous random variables during the theoretical development. Equation (A.2-8) is all that need be recalled to make the translation back to the real world of Monte Carlo analysis.

A.2.2 Joint Distribution Functions

It is often necessary to describe two or more random variables in relation to each other. These situations are handled by treating each of the random variables x_i as an element of a random vector $\underline{x}$. The joint (probability) distribution function is defined as

$$F(\underline{X}) = P(x_1 \leq X_1, x_2 \leq X_2, \ldots, x_n \leq X_N) \qquad \text{(A.2-9)}$$

From the definition, the following limits hold

$$\lim_{\substack{x_1 \to \infty \\ \vdots \\ x_n \to \infty}} F(\underline{x}) = 1 \qquad \text{(A.2-10)}$$

$$\lim_{\substack{x_1 \to \infty \\ \vdots \\ x_{i-1} \to \infty \\ x_{i+1} \to \infty \\ \vdots \\ x_n \to \infty}} F(\underline{x}) = F_i(x_i) \qquad \text{(A.2-11)}$$

where $F_i(x_i)$ is the distribution function of the i^{th} random variable. The joint (probability) density function is defined as

$$\frac{\partial^n}{\partial x_1 \, \partial x_2 \ldots \partial x_n} F(\underline{x}) = f(\underline{x}) \qquad \text{(A.2-12)}$$

A.2.3 Statistical Independence

Random variables are said to be statistically independent whenever the joint distribution functions factors such that

$$F(\underline{x}) = \prod_{i=1}^{n} F_i(x_i) \qquad \text{(A.2-13)}$$

or equivalently

$$f(\underline{x}) = \prod_{i=1}^{n} f_i(x_i) \qquad \text{(A.2-14)}$$

where F_i is the distribution function and f_i is the density function of the random variable x_i. Statistically independent random variables are independent in the sense that the value of one does not affect the value of another, and the definition is equivalent to the one given earlier for statistically independent random events.

A.3 Averages

Averages are of great practical importance. These statistics capture a random variable's pertinent characteristics: its average value and quantitative assessments of randomness. Section A.3.1 discusses the mathematical operation of expectation, which is the means for computing averages. Specific averages (i.e., the moments) are discussed in Section A.3.2. Section A.3.3 discusses an important average used to assess the correlation between two random variables. Confidence limits are discussed in Section A.3.4.

A.3.1 Expectation

If g(x) is any function of the random variable x, the expectation (average value) of g(x) is denoted by $E[g(x)]$. For a discrete random variable

$$E[g(x)] = \sum_{i=1}^{n} g(X_i) f_i \tag{A.3-1}$$

where f_i is the probability of occurrence of X_i. In the case of a continuous random variable

$$E[g(x)] = \int_{-\infty}^{\infty} g(x) f(x) dx \tag{A.3-2}$$

where f is the density function. The development presented here considers only continuous random variables. However, it should be kept in mind that completely analogous formulas exist for all distribution types as illustrated by Eq. (A.3-1).

The expectation of a function of a random variable, as given by Eq. (A.3-2), is a weighted average of the function values with the weights determined by the distribution of the random variable. Expectation is a linear operator, which obeys

$$E\left[\sum_i a_i g_i(x_i)\right] = \sum_i a_i E[g_i(x_i)] \tag{A.3-3}$$

where a_i is any non-random number. The distributive law of multiplication,

$$E\left[\prod_i g_i(x_i)\right] = \prod_i E[g_i(x_i)] \qquad (A.3\text{-}4)$$

holds only if the variables are statistically independent. The relation

$$E[g(x)] = g(E[x]) \qquad (A.3\text{-}5)$$

is generally not true either.

The expectation of a vector valued function $g(\underline{x})$ is defined as

$$E[g(\underline{x})] = \int_{-\infty}^{\infty} g(\underline{x})\, f(\underline{x})\, d\underline{x} \qquad (A.3\text{-}6)$$

where $f(\underline{x})$ is the joint probability density function and integration is with respect to all variables. The expectation operator is a linear operator with respect to vector random variables.

A.3.2 Moments

The quantity $E[x^k]$, is called the k^{th} moment of x

$$E[x^k] = \int_{-\infty}^{\infty} x^k f(x)\, dx \qquad (A.3\text{-}7)$$

In particular with k = 1, $E[x] = \bar{x}$ is the mean value of the variable x. The quantity defined by $E[(x-\bar{x})^k]$ is called the k^{th} central moment of x. Moments and central moments are related. For example, the second central moment given as $E[(x-\bar{x})^2]$ can be expressed in terms of the first two moments

$$E[(x-\bar{x})^2] = E[x^2] - \bar{x}^2 \qquad (A.3\text{-}8)$$

by virtue of the fact that $E[x] = \bar{x}$ and the linear property of the expectation operator.

Several of the low order moments have practical interpretations. By definition, the first central moment is zero. The second central moment given as $E[(x-\bar{x})^2]$ is called the variance of x and is denoted σ^2. It is a measure of dispersion of x about the mean. The square root of the variance is called the standard deviation and denoted by σ.

Based on their frequency of general usage, the mean and variance are by far the most important moments. Occasionally, for non-zero mean random variables a coefficient of variation is defined

$$C_\sigma = \frac{\sigma}{\bar{x}} \tag{A.3-9}$$

Of less frequent usage are the third and fourth central moments. The third central moment provides a measure of the asymmetry of the distribution about the mean value and is referred to as skewness. The fourth central moment is referred to as kurtosis and provides an additional measure of the clustering of the distribution about its mean value.

A.3.3 Correlation

Let x and y be random variables with mean values $\bar{x}$ and $\bar{y}$. The expectation of the product

$$E[(x-\bar{x})(y-\bar{y})] = \iint_{-\infty}^{\infty} (x-\bar{x})(y-\bar{y})\, f(x, y)\, dx\, dy \tag{A.3-10}$$

is called the covariance of x and y, Cov(x, y). If x and y are statistically independent, in accordance with (A.2-4), then

$$\mathrm{Cov}(x, y) = E[x-\bar{x}]\, E[y-\bar{y}] = 0 \tag{A.3-11}$$

The converse is not true. That is, in general, the vanishing of the covariance is not sufficient to insure statistical independence. Whenever the covariance is zero, the variables are said to be "uncorrelated."

Frequently a correlation coefficient is used rather than the covariance. The correlation coefficient is defined as

$$\rho_{xy} = \frac{\mathrm{Cov}(x, y)}{\sigma_x \sigma_y} \tag{A.3-12}$$

where the σ_x and σ_y are the standard deviations of x and y. It can be shown that the correlation coefficient always has values in the range -1 to 1. Again, a non-zero value of the correlation coefficient implies statistical dependence but a vaninshing correlation coefficient does not imply statical independence.

The interrelationship between two random phenomena, x and y, is describable to first order by the correlation coefficient, ρ_{xy}. The correlation coefficient is most clearly understood by an example. One method of assessing the presence of correlation between phenomenon y and phenomenon x would be to plot y versus x. Any clustering of data points in the resulting "scatter diagram" would indicate correlation. The correlation can be quantified by fitting a straight line

$$y = a x \tag{A.3-13}$$

to the data (assuming zero means) using least squares techniques to determine a. The result would be the best linear prediction formula for y given x. This formula is referred to as the regression line. The regression line formula can be written in terms of the standard deviations and correlation coefficient

$$\frac{\hat{y}}{\sigma_y} = \rho_{xy} \frac{x}{\sigma_x} \tag{A.3-14}$$

Equation (A.3-14) shows that the correlation coefficient is the slope of the best-fit straight line when the data has been normalized with respect to the standard deviations. Since both x and y are random, the data will be scattered about the regression line. The scatter can be quantified by calculating the root mean square error

$$\mathrm{rms} = \sqrt{\sum_{i=1}^{n} [y_i - \hat{y}(x_i)]^2} \tag{A.3-15}$$

where i denotes the individual data points. The rms error can be shown to be

$$\text{rms} = \sqrt{1 - \rho_{xy}^2}\ \sigma_y \qquad \text{(A.3-16)}$$

Thus the correlation coefficient is a measure of the scatter about the regression line. If ρ_{xy} is zero, the rms error is equal to the uncertainty in the data itself, σ_y. Thus uncorrelated random variables (i.e., $\rho_{xy} = 0$) are completely random relative to each other and no first order interrelationship exists. Since the rms error must be positive, Eq. (A.3-16) requires

$$-1 \leq \rho_{xy} \leq 1 \qquad \text{(A.3-17)}$$

If $\rho_{xy} = \pm 1$ then the rms error is zero, all the data falls on the regression line, y depends linearly on x, and y is not random relative to x.

A.3.4 Confidence Limits

Confidence limits bound the uncertainty in the value of a random variable. The limits are selected to insure the random variable will fall inside, with a certain level of confidence. Since a random variable x is expected to take on its average value $\bar{x}$, the K-confidence limit, a_k, is defined as the value for which

$$P[|x - \bar{x}| \leq a_k] = K \qquad \text{(A.3-18)}$$

This defines a_k in terms of K and the distribution. The random variable will fall inside the interval

$$\bar{x} - a_k \leq x \leq \bar{x} + a_k \qquad \text{(A.3-19)}$$

with a relative frequency of K.

The Chebyshev inequality

$$P[|x - \bar{x}| \leq h\sigma] \geq 1 - \frac{1}{h^2} \qquad \text{(A.3-20)}$$

is plotted in Figure A-2 along with other representative distribution types. It can be seen to be a conservative lower bound. Substituting Eq. (A.3-20) into (A.3-18) and solving for the confidence limit yields

$$a_k \leq \frac{\sigma}{\sqrt{1-K}} \qquad \text{(A.3-21)}$$

Thus a random variable always has a value within 2σ of its mean 75% of the time and within 3σ of its mean 89% of the time. This illustrates that the standard deviation, σ, quantifies randomness regardless of the distribution.

A.4 Typical Distributions

It is generally difficult to precisely determine the distribution of a random variable. However, it is usually possible to assess characteristic properties which make it possible to select a distribution model. Presented here are four commonly used distribution types.

A.4.1 Uniformly Distributed Random Variable (TYPE = 3)

A uniformly distributed random variable is equally likely to take on any value in the interval $\bar{x}$ - T to $\bar{x}$ + T. The density function is

$$f(x) = \begin{cases} 0 & |x - \bar{x}| > T \\ \frac{1}{2T} & |x - \bar{x}| \leq T \end{cases} \qquad \text{(A.4-1)}$$

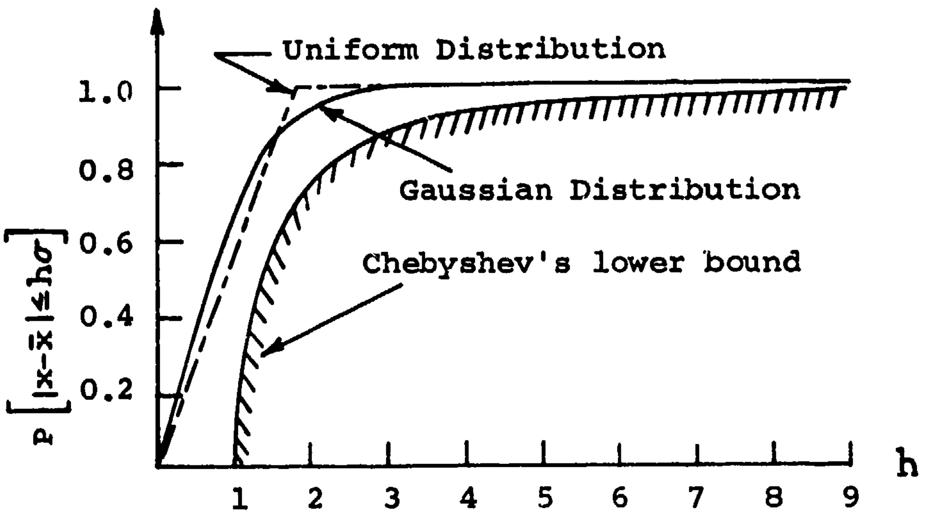


FIGURE A-2   CHEBYSHEV'S LOWER BOUND

The mean value is $\bar{x}$; T is referred to as the tolerance, and

$$\sigma = \frac{T}{\sqrt{3}} \tag{A.4-2}$$

A uniform distribution is characteristic of error sources whose magnitude is limited by quality control procedures. Projectile length is a good example. Uncertainties in projectile length can be easilly controlled by measuring each projectile after final machining. Those exceeding quality control limits would be discarded. Thus, off-nominal lengths beyond these limits do not occur. Within these limits, the length is equally likely to be any value.

A.4.2 Gaussian Distributed Random Variable (TYPE = 2)

A Gaussian distributed random variable is one whose distribution about the mean has the familiar bell shape. The density function is

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2}\left(\frac{x-\bar{x}}{\sigma}\right)^2} \tag{A.4-3}$$

where $\bar{x}$ is the mean value and σ is the standard deviation. Figure A-3 illustrates the Gaussian curve. Since the interval $\bar{x} \pm 3\sigma$ contains better than 99 percent of the outcomes, the tolerance T is defined as

$$T = 3\sigma \tag{A.4-4}$$

The effect of a change in value of σ is indicated in Figure A-4. Increasing σ effects a simultaneous lowering of the peak value and a spreading of the tails of the curve. Changes in the mean value $\bar{x}$ results in a translation of the curve.

The Gaussian form is particularly interesting when sums of random variables are considered. For example, suppose there is a linear functional relation

$$z = ax + by \tag{A.4-5}$$

where a, b are constants and x, y are random variables. The

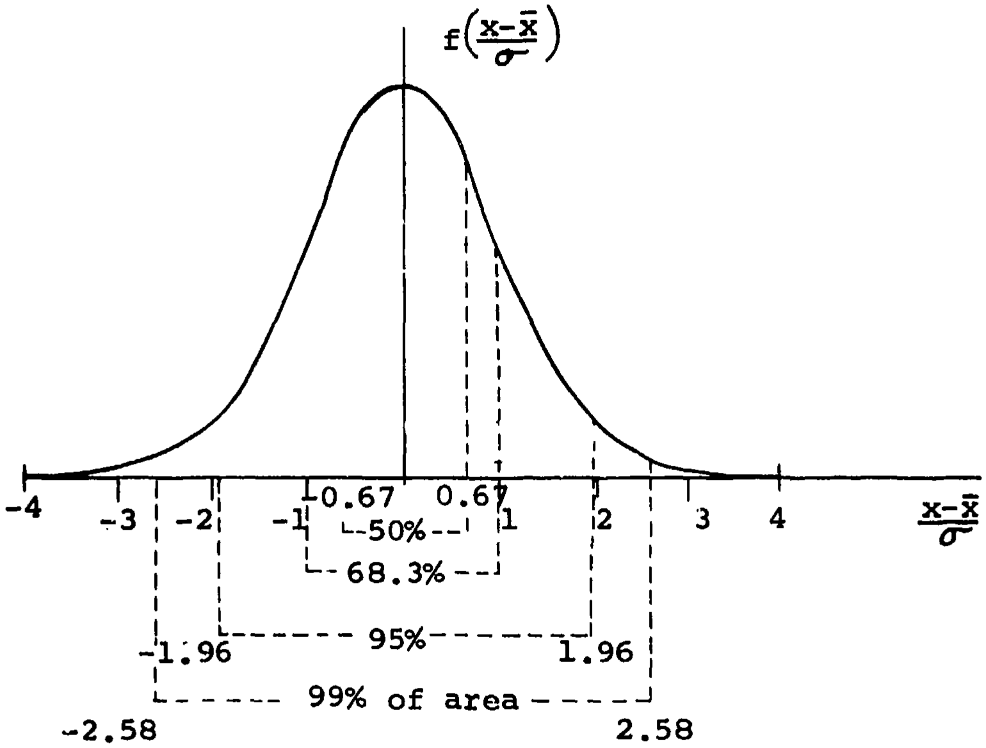


FIGURE A-3 GAUSSIAN DENSITY FUNCTION

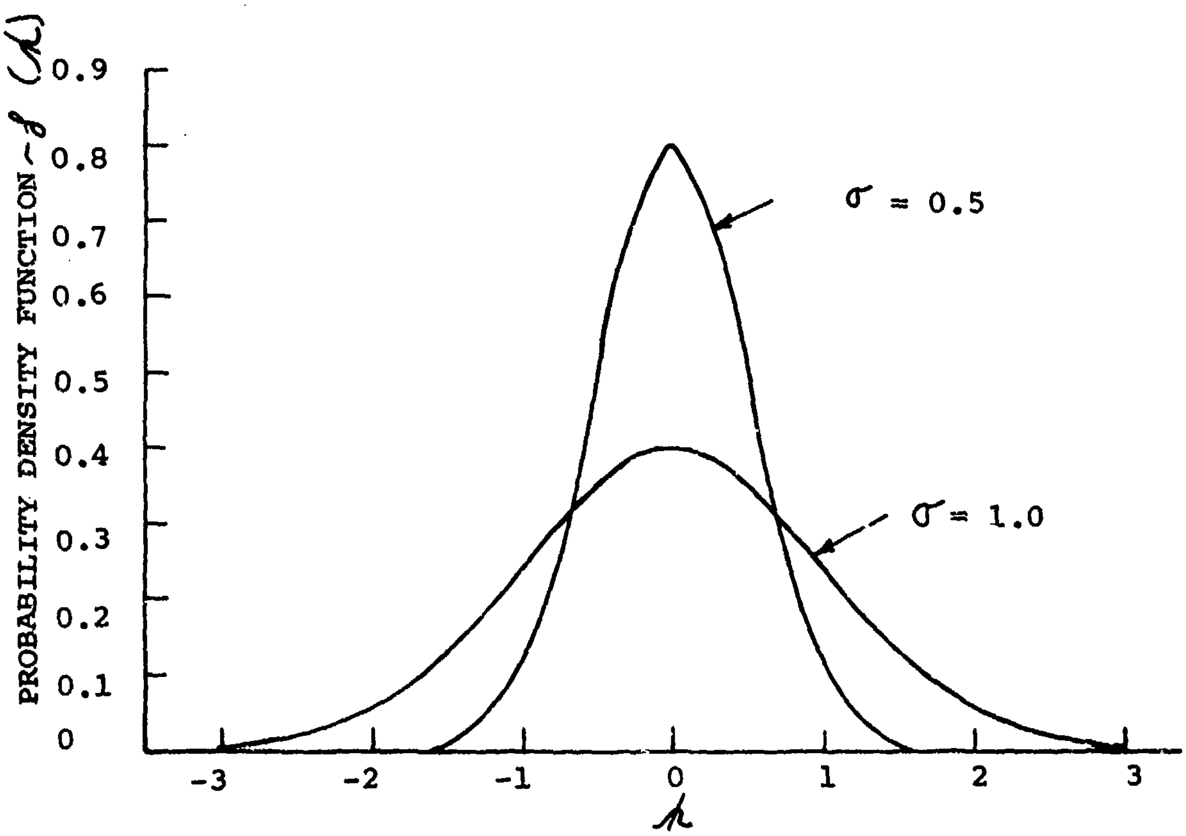


FIGURE A-4 A COMPARISON OF TWO GAUSSIAN DENSITY FUNCTIONS WITH DIFFERENT STANDARD DEVIATIONS variable z, by virtue of being a function of random variables, is also a random variable. If x and y are independent then one can conclude that

$$\bar{z} = a\bar{x} + b\bar{y}$$

$$\sigma_z^2 = a^2 \sigma_x^2 + b^2 \sigma_y^2 \qquad \text{(A.4-6)}$$

by application of the principles previously presented. Now in general, the knowledge of the mean value of z and its variance are insufficient to infer the distribution of z. For example, Figure A-5 illustrates two different density functions which have the same mean and variance. However, it can be shown that if x and y are Gaussian distributed then z will also be Gaussian. This result can be extended to linear combinations of any number of Gaussian random variables.

The Central Limit Theorem states that under suitable conditions the sum of arbitrarily distributed independent random variables will become Gaussian as the number of variables becomes large. The necessary condition is that no single term of the sum can dominate. That is, the variance of any one term must not be of the same order as the sum of the variance of the other terms. For instance, the sum of twenty random variables, which are uniformly distributed between 0 and 1,

$$z = \frac{1}{20} \sum_{i=1}^{20} x_i \qquad \text{(A.4-7)}$$

is very nearly Gaussian.

In summary, a Gaussian uncertainty is one whose distribution about the mean has the familiar bell shape. It is appropriate for effects formed from a large number of independent random occurrences. For instance, variations in muzzle velocity are the result of a large number of independent effects occurring while the projectile traverses the barrel. Thus, the distribution of muzzle velocity would be expected to be Gaussian.

A.4.3 Jointly Gaussian Random Variables

Let x and y be jointly distributed Gaussian random variables. Their joint density function is

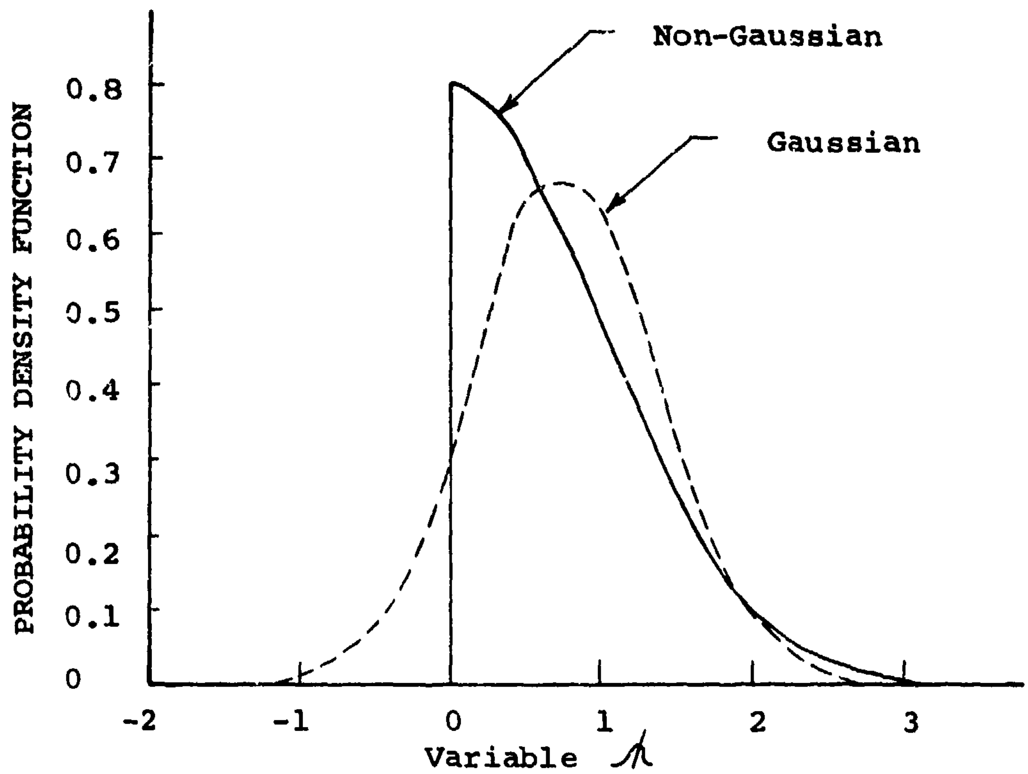


FIGURE A-5 TWO PROBABILITY DENSITY FUNCTIONS WITH IDENTICAL MEANS AND VARIANCES

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y(1-\rho_{xy}^2)^{1/2}} \operatorname{Exp}\left[-\frac{\tilde{x}^2 - 2\rho_{xy}\tilde{x}\tilde{y} + \tilde{y}^2}{2(1-\rho_{xy}^2)}\right] \quad \text{(A.4-8)}$$

where

$$\tilde{x} = \frac{x - \bar{x}}{\sigma_x} \quad \text{(A.4-9)}$$

$$\tilde{y} = \frac{y - \bar{y}}{\sigma_y} \quad \text{(A.4-10)}$$

and $\bar{x}$ and $\bar{y}$ are the mean values of x and y, σ_x and σ_y are the standard deviations of x and y, and ρ_{xy} is the correlation coefficient of x and y. Considered separately, x and y are Gaussian distributed. If the correlation coefficient is zero, the density function factors and x and y are statistically independent. Thus, uncorrelated Gaussian random variables are statistically independent.

Six indicies are commonly used to summarize the randomness of two jointly distributed random variables. The standard deviation, σ, has already been described. There is a 50% probability the point (x, y) will lie between two parallel lines which are equidistant from the origin and are separated by twice the Linear Error Probable (LEP). The Circular Error Probable (CEP) is the radius of the circle with a 50% probability of occurrence,

$$P\left[\sqrt{x^2 + y^2} \leq CEP\right] = 0.5 \quad \text{(A.4-11)}$$

The radius of the 80% circle is denoted R80. The mean radius $\bar{J}$ is the average displacement of the point (x, y) from the origin

$$J = E\left[\sqrt{x^2 + y^2}\right] \quad \text{(A.4-12)}$$

(If x and y are the cross range deflections expressed as a fraction of the range, $\bar{J}$ is referred to as the "average jump angle".) The Radial Standard Deviation (RSD) is the rms value of x and y

$$\text{RSD} = \sqrt{E[x^2 + y^2]} \qquad \text{(A.4-13)}$$

If x and y are jointly distributed Gaussian random variables, a readily available text book[1] gives a detailed development of the analytic expressions relating the CEP to σ_x, σ_y and ρ_{xy}. The CEP is closely approximated by

$$\text{CEP} = 0.589\,(\sigma_x + \sigma_y) \qquad \text{(A.4-14)}$$

when ρ_{xy} is zero and σ_x and σ_y differ by no more than 80% of the larger. Provided σ_x and $\sigma_y = \sigma$ and $\rho_{xy} = 0$, all four indices are proportional to each other as indicated in Figure A-6. Each of these statistics may be used to define a circle which have the probabilities of occurrence as shown in Table A-1.

The above discussion has dealt exclusively with two jointly distributed Gaussian random variables. It is possible to have an arbitrarily large number of jointly distributed Gaussian random variables: $x_1, x_2, \ldots, x_n$. The joint density function is defined elsewhere.[2] The distribution function is completely determined by the NxN covariance matrix

$$\text{Cov}\,(\underline{x}) = \begin{bmatrix} \sigma_{x_1}^{\;2} & \text{Cov}\,(x_1, x_2) & \cdots \\ \text{Cov}\,(x_2, x_1) & \sigma_{x_2}^{\;2} & \cdots \\ \vdots & \vdots & \end{bmatrix} \qquad \text{(A.4-15)}$$

[1]Pitman, G. F., Inertial Guidance, Wiley, New York, 1962

[2]Davenport, N. B., and Root, W. L., An Introduction to the Theory of Random Signals and Noise, McGraw-Hill, New York, 1958.

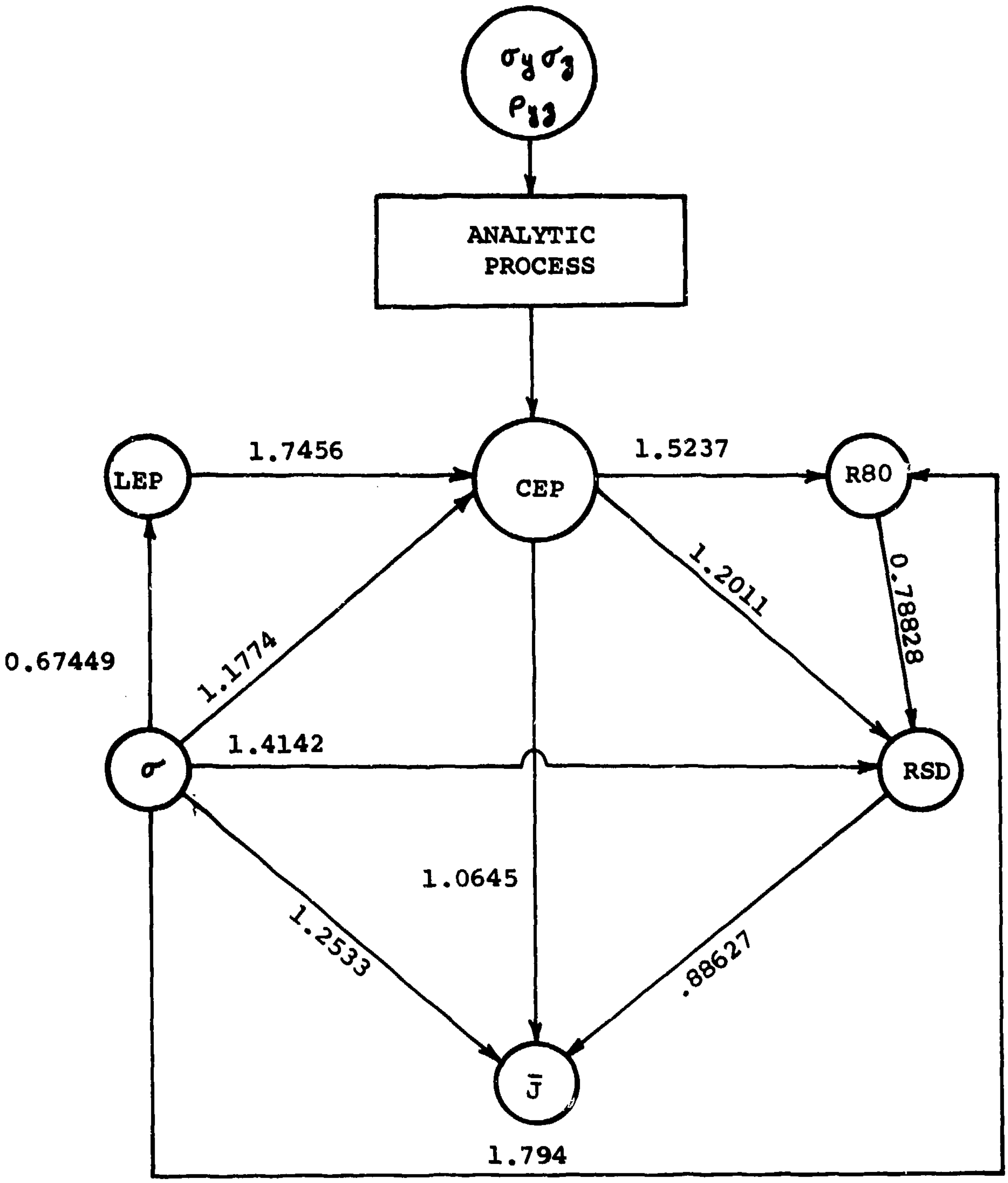


FIGURE A-6 GAUSSIAN SCALE FACTORS

Table A-1 Probabilities of Occurrence

Circle Radius	Probability of Occurrence (%)
LEP	20.3
σ	39.3
CEP	50.0
J	54.4
RSD	63.2
R80	80.0

The covariance matrix is symmetric, positive definite and completely determined by N the standard deviations σ_{x_i} and the correlation coefficients $\rho_{x_i x_j}$.

A.4.4 Rayleigh Distributed Random Variable

Let J be a Rayleigh distributed random variable. The probability density function is

$$f(J) = \frac{\pi}{2} \frac{J}{\bar{J}^2} \text{Exp} \left[- \frac{\pi}{4} \left(\frac{J}{\bar{J}} \right)^2 \right] \tag{A.4-16}$$

where $\bar{J}$ is the mean value of the random variable J. The distribution is completely specified by $\bar{J}$ and

$$\sigma_J = \sqrt{\frac{4}{\pi} - 1}\ \bar{J} = 0.52272\,\bar{J} \tag{A.4-17}$$

The Rayleigh distribution is interesting because of its relationship to two jointly distributed Gaussian random variables. If x and y are identically distributed (i.e., $\sigma_x = \sigma_y = \sigma$) independent (i.e., $\rho_{xy} = 0$) Gaussian random variables, the radius

$$J = \sqrt{x^2 + y^2} \tag{A.4-18}$$

is Rayleigh distributed with mean $\bar{J}$ (i.e, as determined by Figure A-5). The orientation angle

$$\theta = \tan^{-1} y/x \tag{A.4-19}$$

is uniformly distributed between $\pm\pi$ radians. Thus the Rayleigh/uniform distribution describes the Gaussian distribution in polar coordinates.

A.5 Random Variable Generation

In Monte Carlo analyses it is necessary to generate random variables with known distributions. It is desired to make this procedure as simple as possible to conserve running time, simplify conversion to other machines, and not overcomplicate the programming.

The HITS code contains a random number generator which provides uniformly distributed random numbers in the interval from zero to one. This section defines transformations which convert the output of the random number generator into one of three random variable types: uniformly distributed, Gaussian, or arbitrary (tabularly defined) distributed. It is convenient to concurrently describe the method if incrementing the histogram cell counters to record the distribution of the generated random sequence. For this discussion N_c is defined to be the number of histogram cells. R is the uniformly distributed number on the interval zero to one obtained from the random number generator.

A.5.1 Uniformly Distributed Variables (TYPE = 3)

For uniformly distributed variables it is assumed $\overline{x}$ is the mean value and T is the tolerance. The upper and lower bounds defining the extent of the histogram are given by

$$\begin{aligned} x_\ell &= \overline{x} - T \\ x_u &= \overline{x} + T \end{aligned} \tag{A.5-1}$$

The random variable

$$x = \overline{x} + (2R - 1)T \tag{A.5-2}$$

is uniformly distributed on between x_l and x_u. The histogram cell number whose counter should be increased is

$$I = \mathrm{Int}[N_c R] + 1 \tag{A.5-3}$$

where Int yields the largest integer which does not exceed the argument.

A.5.2 Gaussian Distributed Variables (TYPE = 2)

For the Gaussian distributed variables, it is assumed $\overline{x}$ is the mean value and the tolerance T is 3σ where σ is the standard deviation. The upper and lower bounds for the histogram are

$$x_\ell = \overline{x} - 3\sigma \qquad \text{(A.5-4)}$$

$$x_u = \overline{x} + 3\sigma$$

so that the total range of the histograms is 6σ .

A uniformly distributed random number R is converted into a Gaussian random number by the following procedure which uses the conversion table:

r_1 = 0.0	r_6 = 0.50	r_{11} = 1.00
r_2 = 0.28	r_7 = 0.54	
r_3 = 0.36	r_8 = 0.58	
r_4 = 0.41	r_9 = 0.64	
r_5 = 0.46	r_{10} = 0.78	

An index J is calculated according to

$$J = \text{Int}[10\,R] + 1 \qquad \text{(A.5-5)}$$

where the integer function Int has previously been defined. The Gaussian distributed number is then calculated from

$$R_G = (10\,R - J + 1)(r_{J+1} - r_J) + r_J \qquad \text{(A.5-6)}$$

R_G is Gaussian with a mean of ½ and a standard deviation of 1/6. The sample value, R_G, is then scaled to form the sample of the desired Gaussian random variable

$$x = \overline{x} + (2R_G - 1)\,T \qquad \text{(A.5-7)}$$

and the appropriate histogram cell number is given by

$$I = \text{Int}[N_c\,R_G] + 1 \qquad \text{(A.5-8)}$$

A.5.3 Arbitrarily Distributed Variable (TYPE = 4)

For the arbitrarily distributed random variables, the probability density function is defined by a sequence of points x_i, f_i. It is assumed that these values are tabulated with equal divisions of the x coordinate. Furthermore, the average value of the x coordinates is assumed equal to the mean value and occurs midway in the table, the probability of exceeding the x table is zero, and the maximum value for the distribution density function is known. That is

$$\bar{x} = \frac{1}{2}(x_1 + x_k) \qquad \text{(A.5-9)}$$

$$T = \frac{1}{2}(x_k - x_1) \qquad \text{(A.5-10)}$$

$$(f_i)_{max} \text{ is known} \qquad \text{(A.5-11)}$$

Two uniformly distributed random numbers are generated and are designated R_1 and R_2. Using R_1, a tentative sample value is generated from the expression

$$x = \bar{x} + (2R_1 - 1)\,T \qquad \text{(A.5-12)}$$

with an associated histogram interval number being given as

$$I = \text{Int}[N_c R_1] + 1 \qquad \text{(A.5-13)}$$

An index J is also determined from the first random number by the parallel expression

$$J = \text{Int}[K R_1] + 1 \qquad \text{(A.5-14)}$$

where K is the number of tabulated points. The index J is an interpolative index which determines between which two values, x_J and x_{J+1}, R_1 actually lies. Using this index the value of the probability density function is approximated by linear interpolation

$$f(x) = f_J + \frac{(x - x_J)}{x_{J+1} - x_J}(f_{J+1} - f_J) \qquad \text{(A.5-15)}$$

The ratio of this value to the maximal value is calculated and compared with the second random number. If the relation,

$$\frac{f}{(f_i)_{max}} \geq R_2 \qquad \text{(A.5-16)}$$

is satisfied then the sample value of the variable is accepted; if it is not, the sample value is rejected, two more random numbers are generated and the process repeated. This is performed until an acceptable pair of random numbers is found or a predetermined number of trials (20) have been performed in which case a system level error is generated and execution is suspended.

This method is called Von Newmann rejection sampling. The method will generate samples which emulate the given distribution function. However, it can be inefficient. Referring to Figure A-7, the area A_1 represents the area under the distribution function in the transformed plane of the two random numbers. The method consists of accepting pairs of numbers which lie under the transformed distribution density curve.

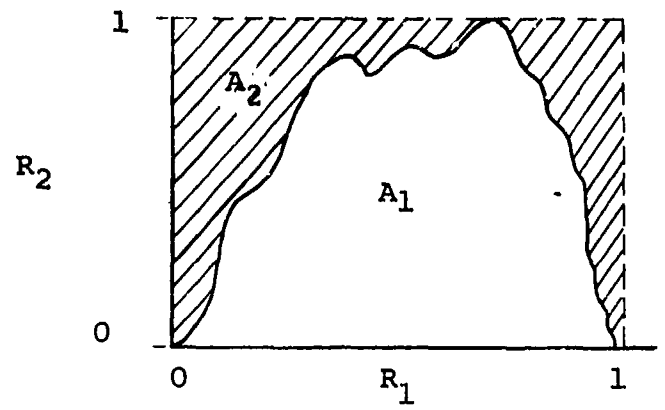


FIGURE A-7 VON NEWMANN REJECTION SAMPLING ILLUSTRATION

The area A_2 represents the region where rejection occurs. A sampling efficiency, η , can be defined as the ratio of sample points accepted to the total number of trials. This efficiency is

$$\eta = \frac{A_1}{A_1 + A_2} = A_1 = \frac{1}{(f_i)_{max}\, 2T} \qquad \text{(A.5-17)}$$

Typically, it can be expected that on the order of 50% of the samples will be rejected, which is an acceptable rate. For very peaked distributions this method could prove to be inefficient. A large number of arbitrarily distributed variables would compound the inefficiency. If this inefficiency proves to be a hinderance, other methods would be more appropriate and should be substituted.

A.6 Monte Carlo Inaccuracies

This section discusses two sources of error in Monte Carlo techniques: the grouping error and the sampling error. Guidelines are developed for minimizing these errors. In summary, these errors are small whenever the histogram cells are small and the sample size is large.

A.6.1 Grouping Errors

In a Monte Carlo analysis, the values resulting from the experiment are sorted into histogram cells and counted rather than recording the precise values. The purpose is to conserve computer storage. Subsequently, the histogram data is used to calculate moments. Since the exact values are lost in the process, an error, termed the grouping error, is introduced in the moments

To illustrate the problem, consider the random variable x with density function f(x). During Monte Carlo experiments the observed values of x are grouped into cells which is tantamount to assuming that all observed values fell at the midpoint of the cell. This is an error. Thus, in reality the histogram does not correspond to the true distribution of x but rather to a discrete distribution where x can take only values associated with the midpoints of the cells. That is

$$x = x_{m_i} \qquad i = 1, \ldots, N_c \tag{A.6-1}$$

are the only possibilities where x_{M_i} are the midpoints. The probability associated with each of these values is

$$P\left[x_{b_i} \leq x \leq x_{b_{i+1}}\right] = \int_{x_{b_i}}^{x_{b_{i+1}}} f(x)\,dx \tag{A.6-2}$$

where x_{bi} and x_{bi+i} are the lower and upper cell boundaries. The ensuing moment calculation based on the histogram is

$$\hat{x}^n = \sum_{i=1}^{N_c} x_{m_i}^n \int_{x_{b_i}}^{x_{b_{i+1}}} f(x)\,dx \qquad \text{(A.6-3)}$$

As a prelude to evaluating the grouping error, consider the true n^{th} moment of x given by the integral

$$\overline{x^n} = \int_{-\infty}^{\infty} x^n f(x)\,dx \qquad \text{(A.6-4)}$$

This integral can be subdivided into the contributions from each histogram cell to yield

$$\overline{x^n} = \sum_{i=1}^{N_c} \int_{x_{b_i}}^{x_{b_{i+1}}} f(x)\,x^n\,dx \qquad \text{(A.6-5)}$$

where x_{bi}, x_{bi+i} are the boundary values of the cells and it is assumed

$$f(x) = 0 \qquad \begin{array}{l} x < x_{b_o} \\ x > x_{b_{N_c}+1} \end{array} \qquad \text{(A.6-6)}$$

The grouping error is the difference between the calculated moment, Eq. (A.6-3), and the true moment as given by Eq. (A.6-5). The exact magnitude of the grouping error in the n^{th} moment depends on the specific distribution. It cannot be evaluated in general unless the density function is approximated. A good first order assumption is all that is required: it is assumed that the density function, f(x), is constant over each histogram cell, i.e.,

$$f(x) = f_i \qquad x_{b_i} < x \leq x_{b_{i+1}} \qquad \text{(A.6-7)}$$

and each of the cells are assumed to be equal in size, $\Delta x = x_{bi+i} - x_{bi}$. Thus, the midpoints of the cells are given by

$$x_{m_i} = x_{b_i} + \frac{\Delta x}{2} \tag{A.6-8}$$

The grouping error, ϵ^N, is the difference between the true and calculated moments of order N

$$\epsilon^n = \overline{x^n} - \hat{x}^n = \sum_{i=1}^{N_c} f_i \left[\int_{x_{b_i}}^{x_{b_{i+1}}} x^n \, dx - x_{m_i}^n \, \Delta x \right] \tag{A.6-9}$$

With the aid of the identity

$$\sum_{i=1}^{n} f_i \, \Delta x = \sum_{i=1}^{n} \int_{x_{b_i}}^{x_{b_{i+1}}} f(x) \, dx = 1 \tag{A.6-10}$$

the grouping error can be evaluated for low order moments. In particular

$$\epsilon^1 = 0$$

$$\epsilon^2 = \frac{\Delta x^2}{12} \tag{A.6-11}$$

Thus, to first order, there is no grouping error in the calculation of the mean value; but, there is an error in the calculated variance equal to $\Delta x^2/12$, where Δx is the cell size.

The expressions for the grouping error given by Eq. (A.6-11) is quite accurate. A more general derivation with a more accurate approximation to the distribution function yields the same result. Corrections for grouping errors are called Sheppards corrections and are treated in greater detail in appropriate texts on statistics.[1] These corrections are not applied in the present Monte Carlo code because they can be made arbitrarily small. Since cell sizes are selected in proportion to the standard deviation of the variables, the percentage error due to grouping is of the

---

[1] Cromer, H., "Mathematical Methods of Statistics," Princeton University Press, 1946.

order $(10/N_C)^2$ where N_C is the number of histogram cells. Thus, using more than ten cells renders the grouping error on the order of one percent of the standard deviation or less, which is small in comparison to other sources of Monte Carlo uncertainties.

A.6.2 Sampling Errors

Monte Carlo methods are exact only in the limit as the number of experiments (or samples) becomes arbitrarily large. Results based on finite sample sizes contain errors called sampling errors, because finite sequences of random numbers are not representative of the entire distribution. This section treats the effects of finite sample size on estimates of the density function and the moments. In summary, sampling errors can be minimized by taking sufficiently large sample sizes. Guidelines are developed.

Estimated Density Function

A random number generator seeks to generate random numbers which are uniformly distributed over the interval zero to one. Let a sequence of N_T such numbers be sorted into N_C histogram cells of equal size covering the range zero to one. It would be expected that each cell would occur N_T/N_C times, or a relative frequency of $1/N_C$. Thus, for $N_C = 10$ and $N_T = 100$, 10 values should fall in each cell and the indicated probability of occurrence of any given cell would be 1/10. However, when an actual sequence of random numbers is sorted it is generally found that neither of these theoretical expectations is true. This section discusses this problem. Only uniformly distributed random variables are discussed, although the results are generally true regardless of distribution type.

The above hypothetical situation can be explained with the aid of the binomial distribution. The binomial distribution states that if an event has a probability p of occurring and a probability $q = 1 - p$ of not occurring, then the probability of exactly X occurrences of the event in N trials is

$$P(X) = \frac{N!}{X!\,(N-X)!}\, p^X q^{N-X} \qquad \text{(A.6-12)}$$

For any particular cell, the probability that the generated random number will fall in that cell is $p = 1/N_C$, with the nonoccurrence probability being $q = 1 - 1/N_C$. According to Eq. (A.6-12), the probability any cell will occur exactly N_T/N_C times in N_T

trials is

$$P\left(\frac{N_T}{N_c}\right) = \frac{N_T!}{\left(\frac{N_T}{N_c}\right)!\left[N_T\left(1 - \frac{N_T}{N_c}\right)\right]!}\left(\frac{1}{N_c}\right)^{\frac{N_T}{N_c}}\left(1 - \frac{1}{N_c}\right)^{N_T\left(1 - \frac{1}{N_c}\right)} \qquad \text{(A.6-13)}$$

where it is assumed that N_T/N_C is an integer quantity. With the assumption that the number of samples is large relative to the number of cells, $N_T >> N_C$, Stirlings approximation

$$N! \approx \sqrt{2\pi N}\, N^N e^{-N} \qquad \text{(A.6-14)}$$

can be used to simplify Eq. (A.6-13):

$$P\left[\frac{N_T}{N_c}\right] \approx \sqrt{\frac{N_c}{2\pi N_T(1 - 1/N_c)}} \qquad \text{(A.6-15)}$$

Using this expression it is found that for a sequence of 100 random numbers sorted into 10 cells the probability any cell occurs exactly 10 times is about 1 chance in 8. Increasing the size of the sample to 1000 doesn't help, the probability of exactly 100 samples in a cell is about 1 chance in 24. Alternately, sorting the sequence of 100 into 5 cells doesn't help either; the probability of any cell occurring exactly 20 times is about 1 chance in 10. This illustrates that as the sample size is increased, or the number of cells is decreased, the probability of observing the theoretically expected number of occurrences per cell diminishes.

It would appear there is no means for obtaining a more exact definition of the probability density function. This is not true, since the density function is more precisely defined in an average sense by larger sample sizes, as illustrated by the following example.

A sequence of 100 random numbers sorted into five cells has an expected number of occurrences of 20 per cell. A 5% deviation in the actual occurrences per cell would be either 19, 20 or 21 in a cell. The associated probability is

$$P(19) + P(20) + P(21) \approx 3\,P(20) = 0.3 \qquad \text{(A.6-16)}$$

Increasing the sequence of random numbers tenfold, the expected number of occurrences is 200 per cell and a 5% deviation would span the twenty values 190, 191,..., 210. The associated probability is

$$\sum_{n=190}^{210} P(n) \approx 20\,P(200) = 0.6 \qquad \text{(A.6-17)}$$

Thus, the chances of being within 5% of the theoretical are about twice as good with 1000 samples as they are for 100 samples. This example illustrates that the number of occurrences tends to the theoretical as the number of trials tends to infinity in an average sort of way. Thus, larger sample sizes improve the estimate of the density function. A general rule of thumb is contained in the old saw, "a few hundred is too few and a thousand is plenty."

Estimated Moments

Monte Carlo methods estimate the moments using Eq. (A.6-3) by approximating the density function with the observed relative frequency

$$f(x) = \frac{N_i}{N_T} \frac{1}{x_{b_{i+1}} - x_{b_i}} \qquad x_{b_i} \leq x \leq x_{b_{i+1}} \qquad \text{(A.6-18)}$$

so that

$$\int_{x_{b_i}}^{x_{b_{i+1}}} f(x)\,dx \approx \frac{N_i}{N_T} \qquad \text{(A.6-19)}$$

where N_i is the number of occurrences of the i^{th} cell, N_T is the total number samples, and x_{b_i} and $x_{b_{i+i}}$ are the lower and upper bounds of the cell. Thus, errors in the estimated moments are the compound effect of grouping errors and density function uncertainties.

Section A.6.1 showed that grouping errors can be made arbitrarily small by using enough cells. This section treats the moment errors associated with uncertainties in the estimated density function. In order to evaluate the errors in the moments determined by Monte Carlo methods, let

$$y = y(x) \tag{A.6-20}$$

where x is the independent random variable and y is dependent. Since x is a random variable and y is a function of x, y is also a random variable. Therefore, y has a density function f_y with moments $\overline{y}$, the mean value, and $\sigma_y{}^2$, the variance, defined by

$$\overline{y} = \int_{-\infty}^{\infty} y\, f(y)\, dy$$

$$\sigma_y{}^2 = \int_{-\infty}^{\infty} (y - \overline{y})^2\, f(y)\, dy \tag{A.6-21}$$

The problem is to assess the errors in the mean value and variance computed from a finite Monte Carlo sequence.

In applying the Monte Carlo method a sequence of random values: $x_1, x_2, \ldots, x_{N_T}$ is generated and the resulting sample sequence $y_1, y_2, \ldots, y_{N_T}$ is computed. This latter sequence is a sequence of random numbers which in turn has a mean

$$\mu_1 = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \tag{A.6-22}$$

If a second sample is generated using a different sequence of random numbers, the second sample will have a mean given as

$$\mu_2 = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \tag{A.6-23}$$

which in general will be different from the first. If the process is repeated many times then a set of values, $\mu_1, \mu_2, \ldots, \mu_J$

will result. This set of values has a distribution function and a mean value and variance associated with it. The question arises then as whether the important properties of the sample mean distribution can be determined without actually determining the sampling distribution itself. This would provide the necessary guidelines for minimizing the sample errors in the moments. This can be done. The sample mean

$$\mu = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \tag{A.6-24}$$

is a function of N_T statistically independent variables; y_i, i = 1, ..., N. Forming the expectation of Eq. (A.6-24) gives the mean value of the sample mean

$$E[\mu] = \frac{1}{N_T} \sum_{i=1}^{N_T} E[y_i] \tag{A.6-25}$$

which, according to Eq. (A.6-21), is

$$E[\mu] = \overline{y} \tag{A.6-26}$$

Therefore, the mean value of the sample mean is equal to the mean. The variance of the sample mean is found similarly. Thus noting

$$\mu - \overline{y} = \frac{1}{N_T} \sum_{i=1}^{N_T} (y_i - \overline{y}) \tag{A.6-27}$$

squaring both sides and forming the expectation

$$\mathrm{Var}(\mu) = E[(\mu - \overline{y})^2] = \frac{1}{N_T} E\left[\sum_{i=1}^{N_T} (y_i - \overline{y})^2\right] \tag{A.6-28}$$

$$\mathrm{Var}(\mu) = \frac{1}{N_T^2} \sum_{i=1}^{N_T} E[(y_i - \overline{y})^2] \tag{A.6-29}$$

Since $E\ (y_i - y)^2 = \sigma_y^2$ the variance of the sample mean is

$$\mathrm{Var}(\mu) = \frac{\sigma_y^2}{N_T} \tag{A.6-30}$$

Thus for a sample size of N_T of a random variable y, the standard deviation of the sample mean is $\sigma_y/\sqrt{N_T}$ where σ_y is the standard deviation of y and N_T is the sample size. On the basis of a single Monte Carlo sample of size N_T, one would report the estimate of the mean value $\overline{y}$ as

$$\overline{y} = \mu \pm \frac{\sigma_y}{\sqrt{N_T}} \tag{A.6-31}$$

The $\sigma_y/\sqrt{N_T}$ is termed a sampling error for the mean. To reduce the sampling error by half requires a quadruple increase in the sample size. By the central limit theorem, for a large number of trials the distribution of the sample mean will tend to be Gaussian. It is therefore possible to associate confidence limits with it. For example, the true value of $\overline{y}$ should lie within $\mu \pm 3\ \sigma_y/\sqrt{N_T}$ with 95% confidence.

Just as there is a distribution of the sample mean, there is also a distribution associated with the sample variance

$$S^2 = \frac{1}{N_T - 1} \sum_{i=1}^{N_T} (y_i - \mu)^2 \tag{A.6-32}$$

The expected value of the sample variance is

$$E[S^2] = \sigma_y^2 \tag{A.6-33}$$

The $(N_T - 1)$ factor in Eq. (A.6-32) is included so the sample variance is unbiased; that is, without an expected error. The standard deviation of the sample variance is

$$\sigma_{S^2} = \sigma_y^2 \sqrt{\frac{2}{N_T - 1}} \tag{A.6-34}$$

To summarize, the moments computed from a Monte Carlo sample of size N_T have the following characteristics

$$\bar{y} = \mu \pm \sigma_y/\sqrt{N_T} \qquad \text{(A.6-35)}$$

$$\sigma^2 = S^2 \pm \sigma_y^2 \sqrt{\frac{2}{N_T - 1}} \qquad \text{(A.6-36)}$$

where μ and S^2 are the sample mean and sample variance, respectively. The HITS code calculates μ and S^2 for each of the variables of interest. The program does not determine the sampling errors involved. The sampling errors are plotted in Figures A-8 and A-9 for various confidence levels. Clearly, samples less than several hundred will produce moments with substantial errors. Samples exceeding a thousand provide adequate accuracy.

While the discussion of the sampling errors in the moments has been based on a single variable, the basic conclusions are applicable to the multiple variable case as long as the variables are statistically independent.

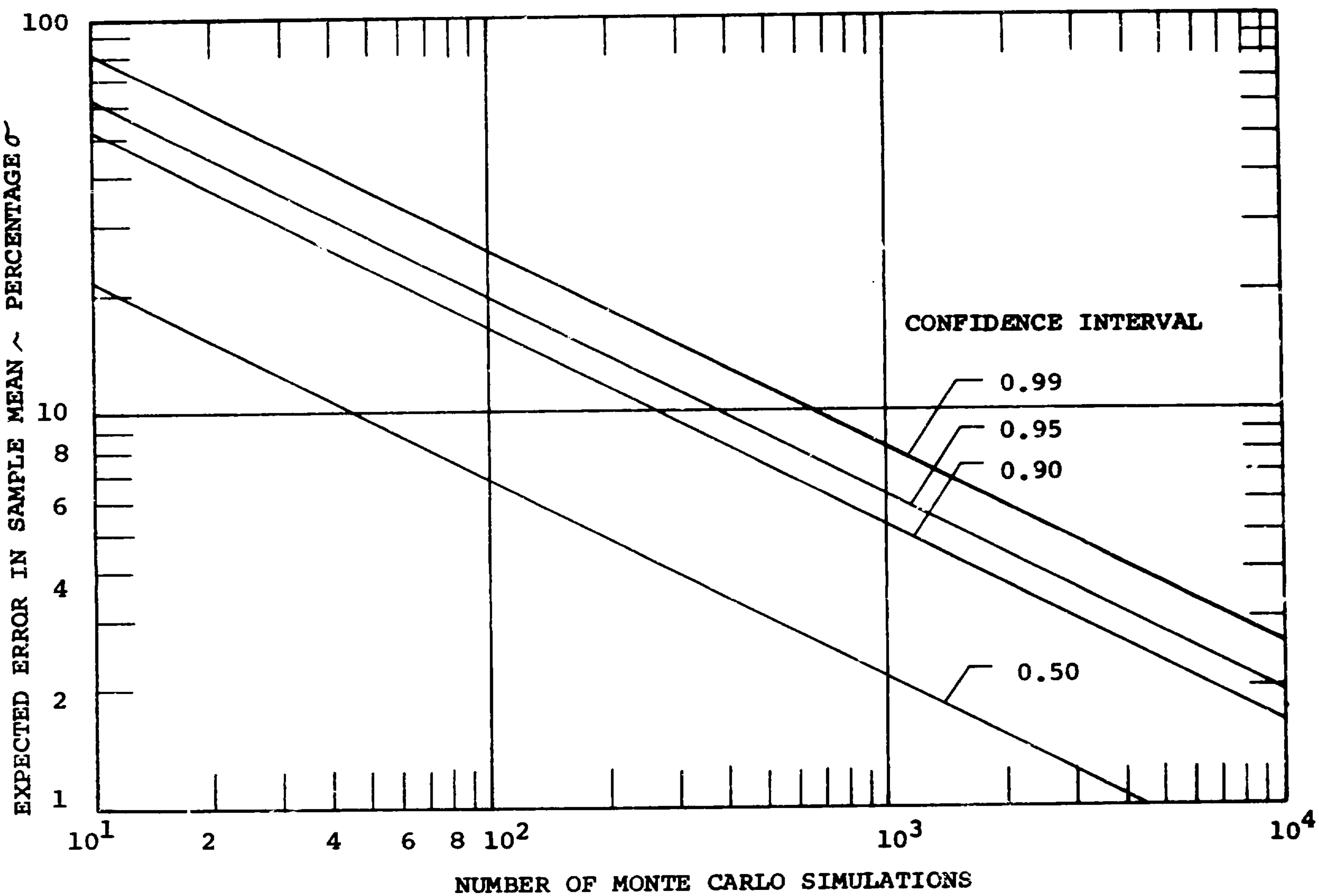


FIGURE A-8 EXPECTED ERROR IN SAMPLE MEAN IN PERCENTAGE OF STANDARD DEVIATION

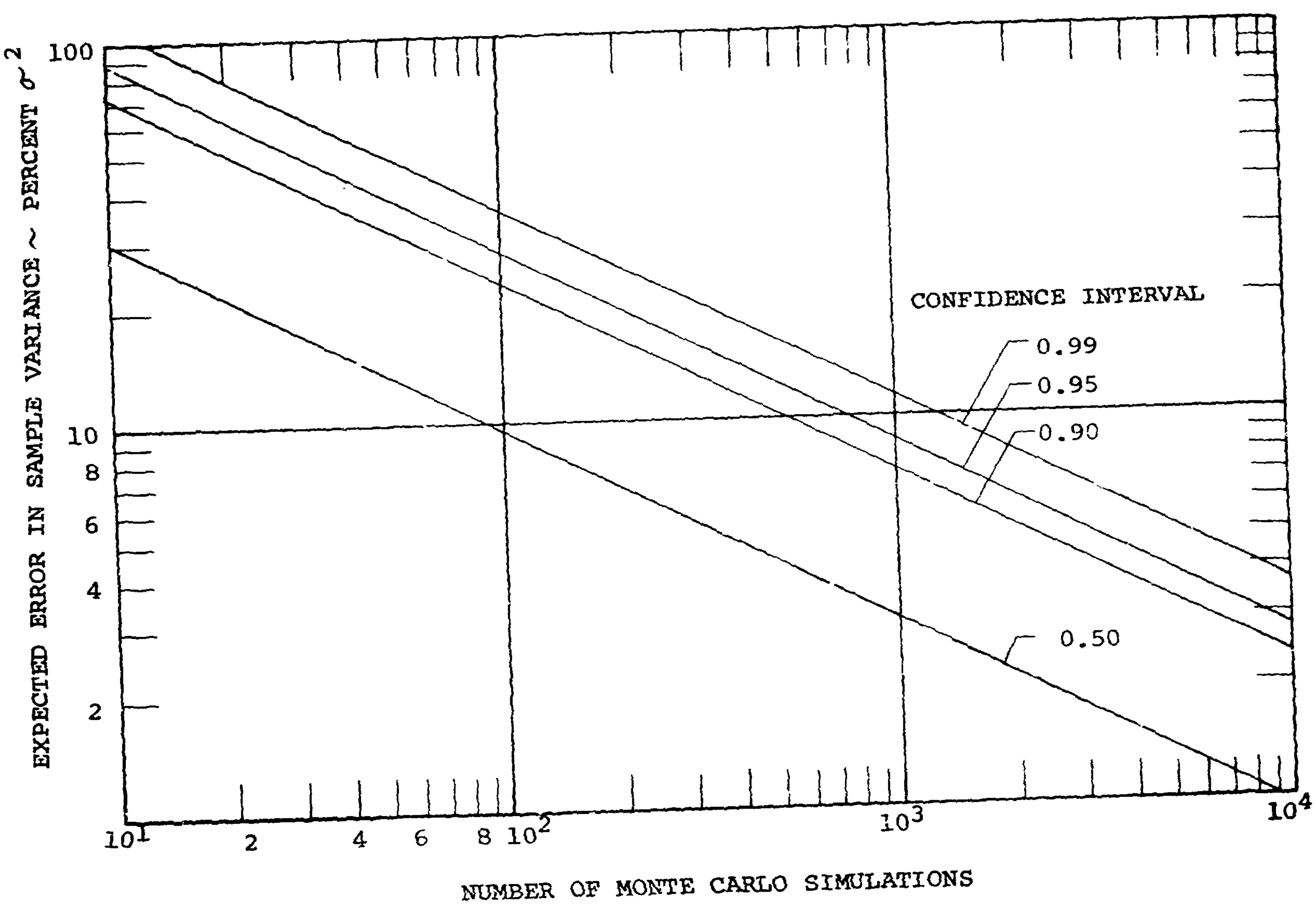


FIGURE A-9 EXPECTED ERROR IN SAMPLE VARIANCE VALUE

to de
Processor
Section B.1.1, and
Section B.1.2.

B.1.1 Analytical Statistical Calculations

The theoretical dispersion assessment problem is to statistically determine the effect of independent variables on a dependent variable. For instance, the effect of the projectile error source model (independent variables) on the down-range dispersion (dependent variable). In general, an exact theoretical solution is not possible. However, the equations can be accurately approximated by a low order Taylor series for reasonable variations in the independent variables, and the mean value and variance of the dependent variable can be estimated. The Analytical Statistical mode mechanizes this concept.

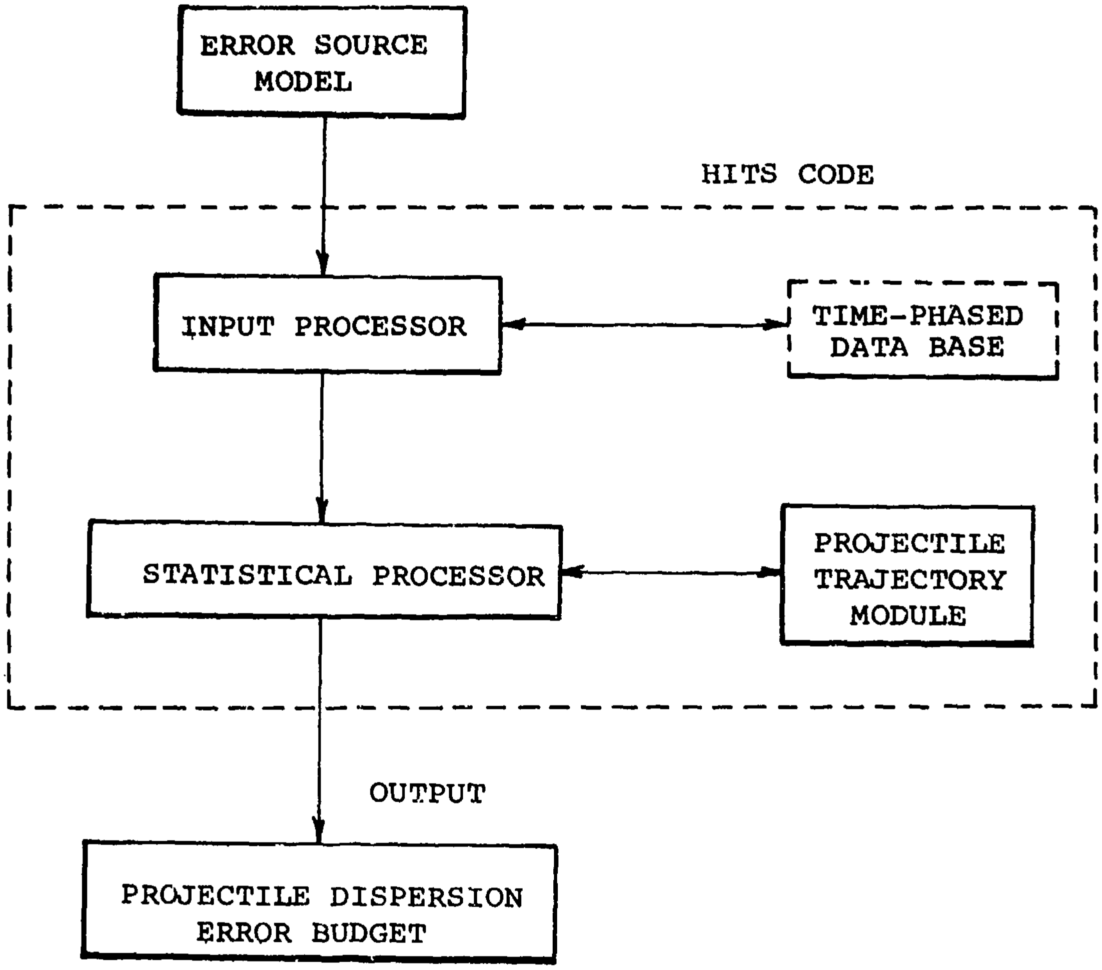


Figure B-1 HITS Flow Diagram

Taylor Series Approximation

The relationship between the dependent variable, y, and the independent variables, $x_1, x_2, \ldots x_n$, is representable as an algebraic equation

$$y(\underline{x}) = y(x_1, x_2, \ldots, x_n) \tag{B.1-1}$$

where $x_1, \ldots x_n$ are treated as the elements of a column vector $\underline{x}$. In general, the equation is non-linear. With the assumption that the function y is differentiable and reasonably well behaved, it can be accurately approximated by a second order Taylor series[1] about the mean value of the independent variables, $\underline{\bar{x}} = (\bar{x}_1, \bar{x}_2, \ldots \bar{x}_n)^T$.

$$y(x) = y(\bar{x}) + \frac{\partial y}{\partial \underline{x}}(\underline{\bar{x}})(x - \bar{x}) + \frac{1}{2}(\underline{x} - \underline{x})^T \frac{\partial}{\partial \underline{x}}\left(\frac{\partial y}{\partial \underline{x}}\right)^T (\underline{\bar{x}})(\underline{x} - \underline{\bar{x}}) \tag{B.1-2}$$

where T denotes the matrix transpose and

$$\frac{\partial y}{\partial \underline{x}}(\underline{\bar{x}}) = \left[\frac{\partial y}{\partial x_1}(\underline{\bar{x}}), \ \frac{\partial y}{\partial x_2}(\underline{\bar{x}}), \ldots, \frac{\partial y}{\partial x_n}(\underline{\bar{x}})\right] \tag{B.1-3}$$

is the gradient row vector of first partial derivatives

[1] Sokolnikoff, I. S., and Sokolnikoff, E. S., Higher Mathematics for Engineers and Physicists, McGraw-Hill, New York, 1941.

and

$$\frac{\partial}{\partial \underline{x}}\left(\frac{\partial y}{\partial \underline{x}}\right)^T(\bar{\underline{x}}) = \begin{bmatrix} \frac{\partial^2 y}{\partial x_1^2}(\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2 \partial x_1}(\bar{\underline{x}}) & \cdots & \frac{\partial^2 y}{\partial x_n \partial x_1}(\bar{\underline{x}}) \\ \frac{\partial^2 y}{\partial x_1 \partial x_2}(\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2^2}(\bar{\underline{x}}) & \cdots & \frac{\partial^2 y}{\partial x_n \partial x_2}(\bar{\underline{x}}) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 y}{\partial x_1 \partial x_n}(\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2 \partial x_n}(\bar{\underline{x}}) & & \frac{\partial^2 y}{\partial x_n^2}(\bar{\underline{x}}) \end{bmatrix} \qquad (B.1\text{-}4)$$

is the Jacobian Matrix of second partial derivatives, both evaluated at the mean value of the independent variables, $\bar{\underline{x}}$.

<u>Dependent Variable Mean Value</u>

The mean value is found by taking the expectation of Eq. (B.1-2). After some manipulation, the mean value of the dependent variable can be shown to be

$$\bar{y} = y(\bar{\underline{x}}) + \frac{1}{2}\sum_{i=1}^{n} \frac{\partial^2 y}{\partial x_i^2}(\bar{\underline{x}})\,\sigma_{x_i}^2 + \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \frac{\partial^2 y}{\partial x_i \partial x_j}(\bar{\underline{x}})\,\mathrm{Cov}(x_i, x_j) \qquad (B.1\text{-}5)$$

This equation states that $\bar{y}$ is composed of the value of y at the mean of $\underline{x}$ plus second order corrections based on the variances and covariances of the independent variables. (Note: HITS includes the second term whenever the control variable IOPRNT $\geq$ 3 and the third whenever IOPRNT $\geq$ 5.)

Dependent Variable Variance

The variance of the dependent variable is computed according to

$$\sigma_y^2 = E[y^2] - \bar{y}^2 \tag{B.1-6}$$

Equations (B.1-2) and (B.1-5) are substituted and manipulated to obtain

$$\begin{aligned}
\sigma_y^2 = {} & \sum_{i=1}^{n} \left[\frac{\partial y}{\partial x_i} (\bar{x}) \right]^2 \sigma_{x_i}^2 + \\
& + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{\partial y}{\partial x_i}(\bar{x}) \frac{\partial y}{\partial x_j}(\bar{x}) \operatorname{Cov}(x_i, x_j) + \\
& + \frac{1}{2} \sum_{i=1}^{n} \left[\frac{\partial^2 y}{\partial x_i^2} (\bar{x}) \right]^2 \sigma_{x_i}^4 + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \right]^2 \sigma_{x_i}^2 \sigma_{x_j}^2 + \\
& + 2 \sum_{j=1}^{n} \sum_{i=1}^{n-1} \sum_{k=i+1}^{n} \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \frac{\partial^2 y}{\partial x_k \partial x_j} (\bar{x}) \right] \sigma_{x_j}^2 \operatorname{Cov}(x_i, x_k) + \\
& + 2 \sum_{j=1}^{n-1} \sum_{m=j+1}^{n} \sum_{i=1}^{n-1} \sum_{k=i+1}^{n} \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \frac{\partial^2 y}{\partial x_k \partial x_m} (\bar{x}) \right] \operatorname{Cov}(x_i, x_k) \operatorname{Cov}(x_j, x_m)
\end{aligned} \tag{B.1-7}$$

Equation (B.1-7) states that σ_y^2 is composed of linear terms based on the variance and covariances of the independent variables and a profuse number of second order corrections. (Note: HITS includes the first term whenever IOPRNT $\geq$ 3 and the second whenever IOPRNT $\geq$ 5.) The second order corrections are exactly correct only when the independent variables are Gaussian, since the development of Eq. (B.1-7) assumed

$$E[(x_i - \bar{x}_i)(x_j - \bar{x}_j)(x_k - \bar{x}_k)(x_m - \bar{x}_m)] =$$

$$\mathrm{Cov}(x_i, x_j)\,\mathrm{Cov}(x_k, x_m) +$$

$$+ \mathrm{Cov}(x_i, x_k)\,\mathrm{Cov}(x_j, x_m) + \mathrm{Cov}(x_i, x_m)\,\mathrm{Cov}(x_j, x_k) \qquad \text{(B.1-8)}$$

(Note: HITS includes the third and fourth terms in Eq. (B.1-7) whenever IOPRNT ≥ 6 and the fifth and sixth whenever IOPRNT ≥ 7.)

Calculation of Derivatives

In order to estimate the mean and variance via Eqs. (B.1-5) and (B.1-7), it is necessary to evaluate the dependent variable and its derivatives at the mean values of the independent variables. The evaluation of $y(\bar{x})$ requires one reference to the projectile trajectory module. HITS determines the derivatives by manipulating the inputs to the projectile trajectory module in a systematic fashion. A central difference scheme is used to numerically approximate the derivatives. The independent variables are incremented one at a time both positively and negatively about their mean values. Letting $\bar{x}_i$ represent the independent variable mean values and Δx_i a positive increment, two function evaluations, (i.e., calls to the projectile trajectory module), are performed to give

$$y^+ = y(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_i + \Delta x_i, \ldots, x_n) \qquad \text{(B.1-9)}$$

$$y^- = y(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_i - \Delta x_i, \ldots, x_n)$$

(Note: HITS sets Δx_i equal to the input tolerance value TØL for each independent variable.) The derivatives are then calculated according to

$$\frac{\partial y}{\partial x_i} = \frac{y^+ - y^-}{2\,\Delta x_i} \qquad \text{(B.1-10)}$$

$$\frac{\partial^2 y}{\partial x_i^2}(\bar{x}) = \frac{y^{+} - 2y(\bar{x}) + y^{-}}{\Delta x_i^2} \qquad \text{(B.1-11)}$$

For the second order mixed derivatives, it is necessary to evaluate the function y four times for each pair of independent variables. The four function evaluations for the pair x_i and x_j are

$$y^{++} = y\left(\bar{x}_1, \ldots, \bar{x}_i + \frac{\Delta x_i}{2}, \ldots, \bar{x}_j + \frac{\Delta x_j}{2}, \ldots, \bar{x}_n\right)$$

$$y^{+-} = y\left(\bar{x}_1, \ldots, \bar{x}_i + \frac{\Delta x_i}{2}, \ldots, \bar{x}_j - \frac{\Delta x_j}{2}, \ldots, \bar{x}_n\right)$$

$$y^{-+} = y\left(\bar{x}_1, \ldots, \bar{x}_i - \frac{\Delta x_i}{2}, \ldots, \bar{x}_j + \frac{\Delta x_j}{2}, \ldots, \bar{x}_n\right) \qquad \text{(B.1-12)}$$

$$y^{--} = y\left(\bar{x}_1, \ldots, \bar{x}_i - \frac{\Delta x_i}{2}, \ldots, \bar{x}_j - \frac{\Delta x_j}{2}, \ldots, \bar{x}_n\right)$$

The second mixed partial derivative is

$$\frac{\partial^2 y}{\partial x_i \partial x_j}(\bar{x}) = \frac{y^{++} + y^{--} - (y^{+-} + y^{-+})}{\Delta x_i \, \Delta x_j} \qquad \text{(B.1-13)}$$

Since the second order mixed partial derivatives are symmetric with respect to the independent variables, there are $n(n-1)$ distinct derivatives for any dependent variable, where n is the number of independent variables.

The increments used in the second order mixed partial derivative calculations are half the values used for the first order derivatives. The reason is illustrated in Figure B-2 which diagrams points in the $x_i - x_j$ plane. An ellipse can be drawn through the four points used for the first derivative calculation. With regards to the assumed Taylor series approximation, the ellipse represents the boundary of the region over which the linearization is valid, by virtue of the selection of the Δx's. For consistency, the off-axis coordinates are halved to insure they will always be interior to the ellipse.

<u>Summary</u>

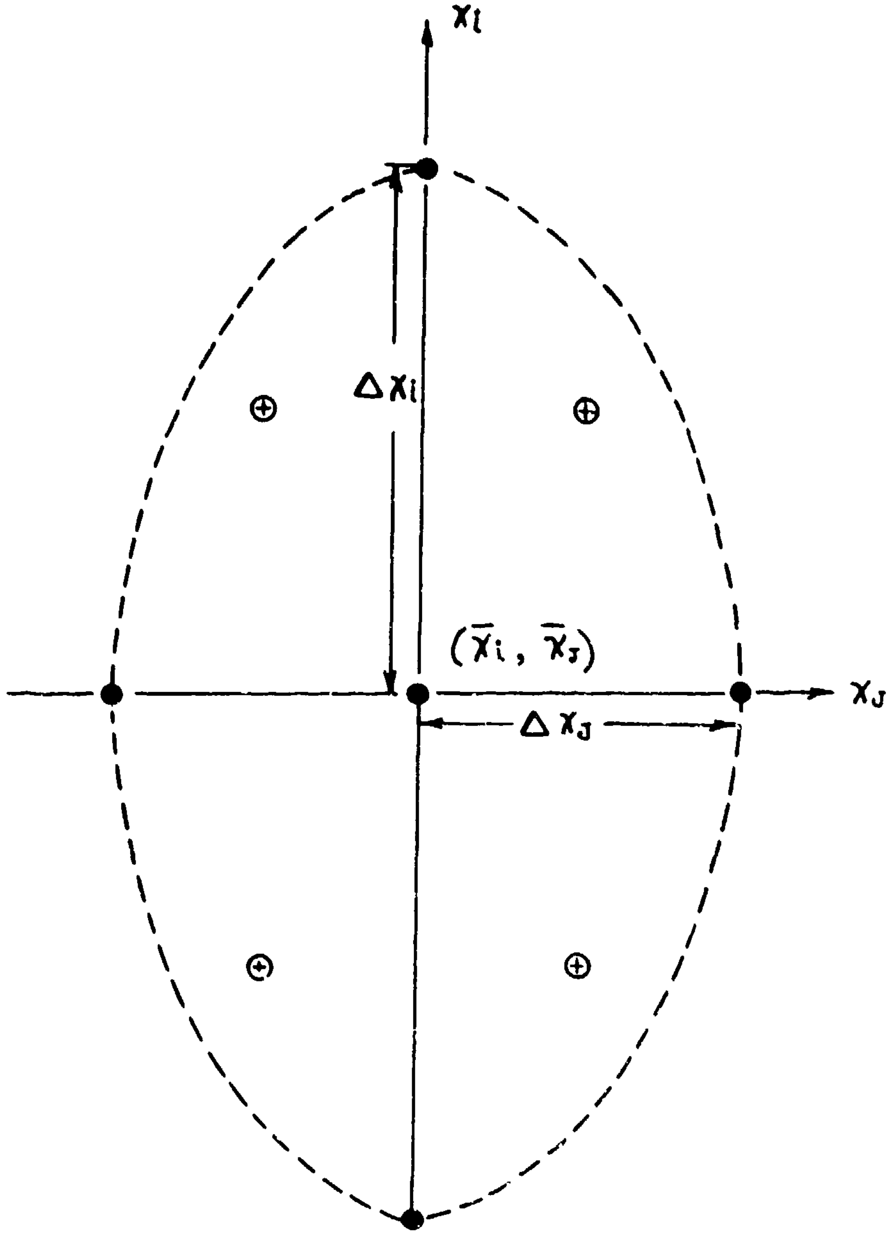
Equations (B.1-5) and (B.1-7) are the basic equations for the Analytical Statistical mode of the Statistical Processor. If there is more than one dependent variable the equations are applied to each in turn. The objective is to determine the mean and variance of all dependent variables from the input statistics of the independent variables. This forces the calculation of first and second order partial derivatives, which is a major undertaking. Since the Analytical Statistical mode is exercised prior to the Monte Carlo mode (to set up the histograms), these calculations are pertinent to the Monte Carlo simulations.

B.1.2 Monte Carlo Calculations

The Monte Carlo mode consists of three parts: (1) generation of representative independent variable values, (2) processing of these values to determine the corresponding dependent variable values, and (3) condensing the ensemble of solutions to a usable form. Representative sequences of the independent variables are obtained from random number generators, as discussed in Appendix A. The trajectory equations of Appendix C relate the dependent variables to the independent. This section discusses the third point: data condensation using histograms.

<u>Cell Definition</u>

In order to achieve substantial condensation, the histograms must be constructed while performing the Monte Carlo experiments. In order to do this, the boundaries of the cells must be determined prior to the experiments. They are determined by an equal division of the range of the variable as defined by the upper bound, x_u, and the lower bound, x_l. The i^{th} cell is $(x_{b_i}, x_{b_{i+i}})$, where the boundaries are given by



● POINTS FOR EVALUATION OF $\frac{\partial}{\partial x_j}, \frac{\partial}{\partial x_i}, \frac{\partial^2}{\partial x_i^2}, \frac{\partial^2}{\partial x_j^2}$

⊕ POINTS FOR EVALUATION OF $\frac{\partial^2}{\partial x_i \partial x_j}$

Figure B-2 Coordinates for Derivative Evaluations

$$x_{b_i} = x_\ell + (i - 1)\,\frac{x_u - x_\ell}{N_c} \tag{B.1-14}$$

where N_c is the number of cells. For independent variables, x_l and x_u are defined in Appendix A.5 for the various different types. For dependent variables, (TYPE = 7), x_l and x_u are determined by the preliminary Analytical Statistical mode calculations. They are

$$\begin{aligned} x_\ell &= \bar{x} - 3\sigma \\ x_u &= \bar{x} + 3\sigma \end{aligned} \tag{B.1-15}$$

where $\bar{x}$ and σ are the calculated values of the mean value and standard deviation. An alternate set of values may be input (TYPE = 8). The cell number for a given value, x, is determined by evaluating

$$i = \mathrm{Int}\left[\frac{x - x_\ell}{x_u - x_\ell}\,N_c\right] + 1 \tag{B.1-16}$$

where Int is the largest integer less than the argument. The midpoint in each cell is termed the "class mark" and is denoted x_{m_i}:

$$x_{m_i} = \frac{1}{2}\left(x_{b_i} + x_{b_{i+1}}\right) = x_\ell + \left(i - \frac{1}{2}\right)\frac{x_u - x_\ell}{N_c} \tag{B.1-17}$$

<u>Density Function Estimate</u>

The Monte Carlo mode counts the number of times each cell occurs during the experimental sequence. The probability density function is estimated according to

$$f(x) = \frac{N_i}{N_T^*}\,\frac{N_c}{x_u - x_\ell} \qquad x_{b_i} \leq x \leq x_{b_{i+1}} \tag{B.1-18}$$

where N_i is the number of times cell "i" occurred and

$$N_T^* = \sum_{i=1}^{N_c} N_i \qquad \text{(B.1-19)}$$

is the total of all cell counts for this variable. Since it is possible for dependent variables to fall outside the range covered by the histogram cells, (x_l, x_u); N_T^* is potentially less than the total number of experiments, N_T.

Moment Estimates

The sample mean and standard deviation are estimated from the histograms. The mean value is

$$\bar{x} = \int_{-\infty}^{\infty} x\, f(x)\, dx = \sum_{i=1}^{N_c} \int_{x_{b_i}}^{x_{b_{i+1}}} x\, f(x)\, dx \qquad \text{(B.1-20)}$$

which becomes the sample mean upon substitution of Eq. (B.1-18),

$$\mu = \sum_{i=1}^{N_c} \frac{N_i}{N_T^*}\, x_{m_i} \qquad \text{(B.1-21)}$$

Using Eq. (B.1-17) the sample mean is

$$\mu = x_\ell + \frac{x_u - x_\ell}{N_T^*\, N_c} \sum_{i=1}^{N_c} \left(i - \frac{1}{2}\right) N_i \qquad \text{(B.1-22)}$$

The variance is defined by

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \bar{x})^2\, f(x)\, dx \qquad \text{(B.1-23)}$$

Upon substitution of Eq. (B.1-18), and using the sample mean instead of the mean value, Eq. (B.1-23) becomes an expression for the sample variance:

$$S^2 = \left(\frac{x_u - x_\ell}{N_c}\right)^2 \left[\sum_{i=1}^{N_c} \left(i - \frac{1}{2}\right)^2 \frac{N_i}{N_T^*} - \left(\sum_{i=1}^{N_c} \left(i - \frac{1}{2}\right) \frac{N_i}{N_T^*} \right)^2 \right] \qquad \text{(B.1-24)}$$

B.2 Code Architecture

This section discusses the architecture of the HITS code. The architecture was designed to meet the requirements of the Statistical Processor computational procedures just described. The four storage arrays discussed in Section B.2.1 form the basic structure. The input processor of Figure B-1 plays the intimate role described in B.2.2. Statistical Processor functions are presented in B.2.3. Subroutine definitions and linking are detailed in B.2.4. The facility for computer generated histogram plots is discussed in B.2.5.

B.2.1 Data Storage

The entire code is designed around four arrays in common storage. These arrays are

1) "OE" array - This one-dimensional array is a common input-output storage area to be shared by the projectile trajectory module and the Statistical Processor. That is, all subroutines are written with all their variables equivalenced to the OE array. The variable "code numbers" are addresses in the OE array.

2) "IA" array - This bicolumned array is used to store address links between the "OE" array and the additional storage areas of the B and C arrays described next. The first 10 rows of the IA array are reserved for dependent variables.

3) "B" array - This one-dimensional array is used to store most input information and some additional values.

4) "C" array - This one-dimensional array is used to store some input information and intermediate output information. It is a general scratch pad storage for the Statistical Processor.

The data transfers between the arrays and the computational modules that effect these transfers are indicated in Figure B-3.

B.2.2 Input Processor Functions

The Input Processor is a collection of subroutines. It determines the kind of calculations to be performed, the subroutines to be used, and the data required. The Input Processor reads the data, checks it for completeness and takes appropriate action if errors are detected or data is found to be missing. If a complete set of data has been input, control is relinquished to the Statistical Processor. This section discusses the functions of the Input Processor in detail.

The Input Processor determines which trajectory module is going to be exercised (at present, there is only the one described in Appendix C, which is specified by IY = [illegible]), clears the IA array, and stores a list of code numbers into the first column of the IA array starting at row 11. This is a complete list of all the addresses in the OE array for which data is required. The first 10 rows of IA are reserved for variables in the OE array which are defined by the input to be dependent variables. The OE array is initialized at preset (i.e., default) values.

The input cards are read and processed one at a time. Generally, the card contains a code number and five values as described in the text. The five values are placed in the B array with the aid of a counter. The first column of the IA array is searched to find the row containing the code number. When it has been located, the current value of the counter is stored in the second column; and the five values are placed in the corresponding location in the B array. Subsequently, the counter is advanced by six and the next data card is read and processed in a like fashion. The input proceeds until a card having a negative code number is encountered which signifies that all data has been read in. After all the data cards have been read, each row of the IA array has two addresses, the first is a location in the OE array, (i.e., the code number), and the second is a location in the B array where data is stored. Input data is extracted by searching the first column of the IA array, and then using the second column to vector into the B array. Thus, the order of the data cards in immaterial. Note: if any rows of the IA array are interchanged the information is not disturbed.

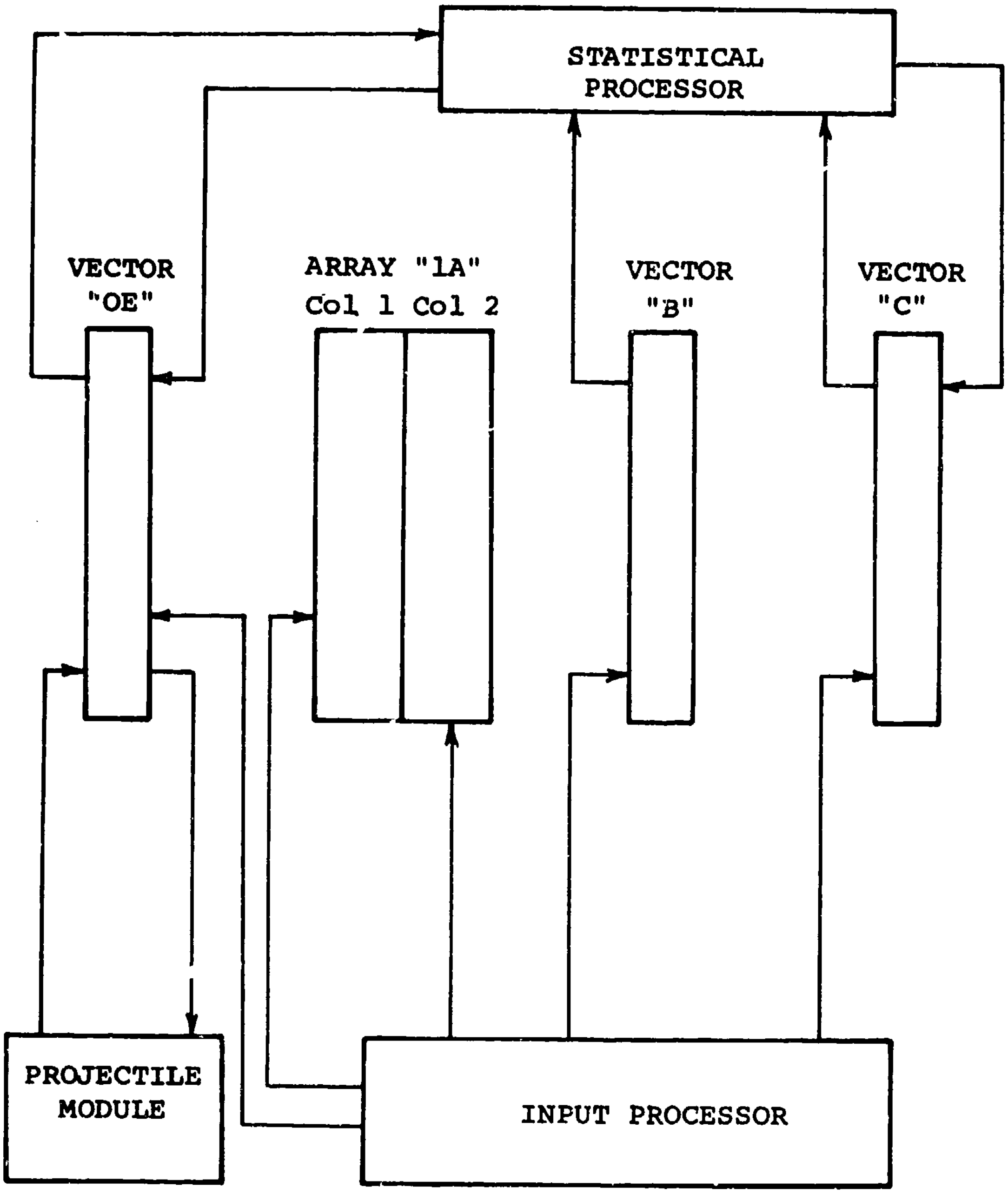


Figure B-3 Internal Data Transfers

If necessary data is missing, zeros will appear in the second column of the IA array. The second column of IA is checked for zeros with appropriate action taken whenever a zero is encountered. At present, a preset value is substituted. However, HITS has the facilities to access a "Time Phased Data Base." This option could be activated if real-time interactive terminals are available.

This description of the Input Processor is valid though over simplified. Several additions and/or modifications are necessary:

1) At the time each data card is read and processed, the variable value on the card is stored in both the OE array and B array. This saves a separate pass through the data to initialize the OE array.

2) For missing data, a "Time Phased Data Base" would be treated like a second source of data. That is, when the card reader is exhausted and data is found missing, the time phase data base is assumed to be implemented as a subroutine which can provide the missing data. This logic is present in the program, but for the present all missing data is filled in with preset values.

3) Dependent variable (TYPE = 7 or 8) code numbers are stored sequentially in the first 10 rows of the IA array. For a TYPE = 7, nothing is stored in the B array, and the second column of the IA array is not disturbed. For a TYPE = 8, the data is stored as previously described.

4) For a constant input value (TYPE = 5), a "-1" is inserted in the second column of the proper row of the IA array to signify that a value was read in. The value is then placed in the OE array.

5) A TYPE = 1 variable has a list of values associated with it. The list is stored in the C array. The TYPE, number of values, and the starting address for the list in the C array are stored sequentially in the B array. The appropriate index of the B array is stored in the IA array, in the appropriate row of the second column.

Since no nominal value is defined for this kind of variable, the first comment does not apply.

6) TYPE = 4 variables have a two-dimensional list associated with them. They are handled like a TYPE = 1 variable (see comment five). However, after read in, a nominal value, tolerance, variance, etc. are calculated and these values are stored in the B array as though they had been read in.

7) Inputs to the projectile trajectory module can be either floating point or fixed point. All input values are read in floating point format. To signify a value has to be stored in the OE array in an integer format, the negative of its code number is stored in the first column of the IA array. The code number is supplied by the Input Processor at initialization and therefore, there is no apparent difference to the user. Also note that integers and floating point numbers are stored in the B and OE arrays with mixed formats. The method is important to understanding the details of the code. Two equivalenced names are used, for instance, say U(5000) and IU(5000,2). The first is double precision floating and the second is single precision integer. When storing an integer value IU (*, 1) is referenced, and when storing a floating number U (*) is referenced.

B.2.3 Statistical Processor Functions

This section describes the control functions performed by the Statistical Processor. There are four modes of operation: Single Trajectory, Range Check, Analytical Statistical, and Monte Carlo.

Single Trajectory

After all data has been read by the Input Processor, the nominal case is executed and the results are stored in the C array. If the input stream contained only TYPE = 5 (constant) variables, the nominal case values are printed and a normal exit results.

Range Check

After all data has been read and all missing values filled in, the rows in the IA array are sorted so that TYPE = 1 (Range Check) variables occupy rows 11, 12 and 13. If there are TYPE = 1 variables, they are processed in a nested DO loop and the values of the dependent variables for each combination are stored in the C array. After the DO loops have been satisfied, the C array information is printed and a normal exit results.

Analytical Statistical

If no TYPE = 1 variables are input, the IA array rows are sorted such that all TYPE = 2, 3 and 4 variables are situated at the top. After this sort, the nominal case is executed. The nominal case results are stored in the C array.

Following the nominal case, limit cases are run with the independent variables sequentially permuted. The resulting dependent variable values are stored in the C array behind the nominal case results. The derivatives are calculated sequentially. The resulting derivative values are stored in the C array by overwriting the previous function evaluations. Using the derivatives, the function mean values and variances are calculated and stored in the C array immediately after the derivatives. The derivatives, means and variances are then printed.

These processes can be time consuming and require large amounts of computer storage. For purposes of estimating the storage and time requirements, let N_D be the number of dependent variables (TYPE = 7 or 8), and N_I be the number of independent variables (TYPE = 2, 3 or 4). As a general rule, storage requirements vary with the product $N_D N_I$, whereas the time consumption varies with N_I. Table B-1 states time (passes through the projectile trajectory module) and storage requirements. The C array allocates a total of 5,000 locations to data storage of the outputs as well as tabular storage for TYPE = 4 variables. This should be more than adequate for most purposes. Should more storage be required, it is a straightforward process to increase the size of the C array in each subroutine in which it appears.

Table B-1 Time and Storage Requirements

Control Variable		Function Evaluations	Required Storage Locations (C Array)
	Nominal Case	1	N_D
IØPRNT ≥ 1	Limit Cases	$2N_I$	$2N_D N_I$
IØPRNT ≥ 2	Derivatives	0	$2N_D$
IØPRNT ≥ 4	Mixed Derivatives	$4N_I(N_I + 1)$	$N_D N_I(N_I-1)/2$
IØPRNT ≥ 5	Covariances	0	$N_I(N_I+1)/2$

Monte Carlo

The Monte Carlo mode performes four functions: It reads additional input, verifies the input, conducts Monte Carlo experiments and constructs histograms, and outputs the results. The input step of the Monte Carlo mode is preceded by execution of the Analytical Statistical mode. The Monte Carlo mode acquires additional control parameters from the input which define the sample size, number of cells, calculational option, and allowable number of rejected trial solutions.

The second function, input verification, consists of rearranging some of the values generated by the Analytical Statistical mode to be compatible with the Monte Carlo calculations. To efficiently describe the verification process, the range of all variables is assumed to be known in terms of

$$\bar{x} - T \leq x \leq \bar{x} + T \tag{B.2-1}$$

where $\bar{x}$ is the mean value and T is the tolerance. The relationship of these variables to inputted and computed quantities has been stated elsewhere. These are summarized here along with some additions and/or modifications. In general for TYPE = 2, 3 or 8 variables the user specifies $\bar{x}$, T, and the variance σ^2, by input. The Analytical Statistical mode calculates these same quantities for TYPE = 4 and 7 variables. To insure proper sampling and to detect possible user errors the input is verified according to the following rules.

1) For TYPE = 2 or 7 variables the tolerance T is taken to be the larger of the input specified value or 3σ and the input value of $\bar{x}$ is accepted.

2) For TYPE = 3 or 8 variables the given tolerance value and input value of $\bar{x}$ are accepted and the variance value is ignored.

3) For TYPE = 4 variables the midrange value and tolerance are calculated from the extremal values of the input list with the input values ignored. For example, if ten values are entered

$$\bar{x} = \frac{1}{2}(x_{10} + x_1) \tag{B.2-2}$$

$$T = \frac{1}{2}(x_{10} - x_1) \tag{B.2-3}$$

For TYPE = 4 variables the maximum value of the distribution function is also found and stored in the location reserved for the calculated variance value. Any quantities changed in the verification process are stored and are subsequently printed on the output. The input verification step also includes the initialization of the histogram counters as well as the counting of the number of TYPE = 8 variables to be considered.

The third function in the computational process is the conduct of the Monte Carlo experiment and the construction of histograms. This is the heart of the whole computation. The independent variable values are generated, processed through the projectile trajectory module, and analyzed to increment appropriate histogram cell counters. The process is then repeated over and over until one of the following terminating conditions occurs:

1) Sampling difficulties for a TYPE = 4 variable occur.

2) The number of trial solutions which are outside of the internally generated histogram ranges exceeds a user defined limit.

3) The number of trial solutions which are outside of the user supplied histogram ranges exceeds a user defined limit.

4) The specified number of experiments have been performed.

Histograms are constructed for both the dependent and independent variables. Reasonable estimates of the ranges of all variables are available from input or are internally computed and cell boundaries are established. During the execution of the Monte Carlo experiments it is possible for one or more dependent variables to fall outside the range covered by its histogram. The results from such an experiment should be rejected from all histograms. This requires the code to take the special precautions described in the next two paragraphs when incrementing the histogram counters.

To accommodate these problems the following scheme is employed. The basic histograms are stored in five integer arrays. Two of these arrays, ID1 and II1, are used to store the count for each cell belonging to the dependent and independent variables, respectively. For example, if each variable range contains five cells, the counter for the second cell of the third dependent variable is the twelfth location of ID1. The arrays ID1 and II1 allow for a maximum of 20 cells with a maximum of 10 dependent variables and 20 independent variables. Two additional integer arrays, IDT(10) and IIT(20), are used for temporary storage of appropriate counter addresses in the ID1 and II1 arrays. As the independent variable values are generated for each trial solution, the appropriate counter addresses are stored in the IIT array. After the dependent variables have been evaluated, their counter addresses are stored in the IDT array. If all of dependent variable values are within the range of their respective histograms, the addresses in the IDT and IIT arrays are used to increment the appropriate counters in the ID1 and II1 arrays. If one or more of the dependent variable values is not within range, then this trial solution is discarded and none of the interval counters are incremented. In this fashion the histograms count only acceptable experiments. A fifth integer array, IR1, is used to count the frequency with which each of the dependent variables causes experiments to be discarded.

The user may desire histogram ranges for the dependent variables which differ from those internally computed. To accommodate this possibility three additional integer arrays, ID2, II2 and IR2, are included. These arrays are replicas of the basic arrays, ID1, II1 and IR1, and are manipulated in exactly the same manner described above with one exception. The exception is that user specifications in the form of TYPE = 8 variables are used instead of the internally generated ranges. Thus, if the user specifies a TYPE = 8 dependent variable, the routine provides two sets of histograms, one set using internal ranges and one set using the TYPE = 8 specifications.

The fourth and final function performed on the Monte Carlo mode consists of printing the Monte Carlo results. The histogram values are calculated and printed. The sample mean and variance are calculated and printed along with the corresponding values from the Analytical Statistical mode. The output is on a variable by variable basis. Occasionally the program hangs up in the output section with an indicated error of a divide fault. This usually occurs because none of the generated solutions were within the histogram ranges. This is usually due to either a gross input error, or an over restricted range for a TYPE = 8 variable.

B.2.4 HITS Subroutines

The HITS computer code consists of a MAIN program and sixty-two subroutines. This section briefly describes the subroutines and their functional interconnections.

Table B-2 lists and briefly describes each of the subroutines. Further details can be gained from the program listings of Appendix D.

In Figure B-4 is a diagram depicting the subroutine linking arrangement of the HITS code. Referring to Figure B-4, there are four decision blocks labeled D1 through D4, each of which has several branch paths. The paths from the decision blocks are mutually exclusive and whenever the code is exercised only one path is followed. Decision block D1 is used to select the projectile trajectory module. At present, there is only one option (IY = 1); the others are dummy facilities. The second decision block, D2, determines which of the four Statistical Processor modes are to be exercised: Single Trajectory, Range Check, Analytical Statistical, or Monte Carlo. Decision block D2 is controlled by the TYPE input data. The decision block D3 is under direct control of the user and affects the decision to use a Taylor series approximation of the Projectile Trajectory module for the Monte Carlo experiments. Decision block D4 determines whether separate histograms are to be generated for user specified histogram ranges. This decision is made on the basis of whether or not any TYPE = 8 variables were input.

Figure B-5 diagrams the subroutine linking arrangement for the Projectile Trajectory module. Note the two distinct trajectory calculations based on the real world and fire control projectile characteristics.

B.2.5 Histogram Computer Plots

The HITS computer code is designed to facilitate creation of computer drawn histogram plots. Since computer plot software is hardware specific, the present code doesn't do the plotting. Rather it gathers the data together into one subroutine (HYSPLT) so that automatic plotting can be easily implemented by users who desire it. This section defines the format of the data.

The two-dimensional array TITLE (I,J) contains the titles for the histograms. The second subscript, J, denotes the variable

Table B-2 HITS Subroutine List

SUBROUTINE	FUNCTION
MAIN	Print input data and sequence other subroutine calls.
ARTLU	One-dimensional table look up routine.
A2	Computes $\int_0^x a^2\,dx$ using Real World parameters.
A2F	Computes $\int_0^x a^2\,dx$ using Fire Control parameters.
NAMES	Block data routine to initialize output variable names.
CHCKIA	Perform check to see if any data is missing.
CHCKIN	Check the input variables: a) establish number of TYPE = 8 variables b) force tolerance = 3σ for TYPE = 2 and 7 variables c) establish tolerance for TYPE = 4 variables.
CNVRT	Convert uniform random numbers to Gaussian random numbers with the interval 0 to 1 as 3σ limits.
CRØSS	Computes the Real World projectile's oscillatory motion and crossrange (horizontal and vertical) velocity and position due to oscillatory motion.
CRØSSF	Computes the Fire Control projectile's oscillatory motion and crossrange (horizontal and vertical) velocity and position due to oscillatory motion.
CRVFT	Generate output variable values using 1st or 2nd order Taylor series expansions of projectile trajectory module.
DØABC	Provide automatic Range Check calculations.
DØ234	Provide Analytical Statistical mode calculations.
EXTRA	Output routine for Range Check variable input data.
FC2987	Fire Control trajectory module interface.

Table B-2 HITS Subroutine List (Cont'd)

SUBROUTINE	FUNCTION
FEXP	Computes double precision exponents with error checking.
FILLA	Calculate mean values and variances from histograms.
FILLIN	Dummy routine to provide interface with time phased data base for missing information.
GQCØR	Computes fifth and sixth terms in Eq. (B.1-7).
GQUC	Computes third and fourth terms in Eq. (B.1-7).
G1795	Dummy alternate trajectory module interface.
G2440	Dummy alternate trajectory module interface.
G2987	Trajectory module interface.
HSTG1	Increment histogram counters using internally estimated variable range values.
HSTG2	Increment histogram counters using user defined (TYPE = 8) variable range values.
HYSPLT	Interface for automatic histogram bar chart plots.
INCARD	Reads data cards and decodes the TYPE numbers
INCØN	Computes algebraic constants using Real World initial conditions.
INCØNF	Computes algebraic constants using Fire Control initial conditions.
INITIL	Selects projectile trajectory module, presets data values and initialize INCARD.
INLHST	Initialize all histogram and error counters to zero.
LØADER	Collects and stores data for automatic histogram bar chart plotter.

Table B-2 HITS Subroutine List (Cont'd)

SUBROUTINE	FUNCTION
MCRL	Sequence Monte Carlo operations and monitor number of trials and number of rejects.
MØVEUP	Internally sorts input data to move variables to top of storage arrays.
NCØV	Locates covariances of independent variables in the C array.
NMEAN	Locates mean values of dependent variables in the C array.
NVAR	Locates variances of dependent variables in the C array.
NVARX	Locates variances of independent variables in B array.
N1D	Locates first order derivatives in C array.
N2D	Locates second order derivatives in C array.
PICK1	Branch routine to select projectile trajectory module.
PRNTMC	Provides printed output of the Monte Carlo results.
QCØR	Computes second term of Eq. (B.1-5) and second term of Eq. (B.1-7).
RANDØM	Formats random numbers.
RANDU	Random number generator for supplying uniform random numbers on interval 0 to 1.
RW2987	Real World trajectory module interface.
SAMPLE	Generates random sample values for all TYPE = 2, 3 or 4 variables.
SPRNT	Formats and prints contents of OE array.
STDDEV	Prints standard deviations with greater precision.
STØREC	Data storage routine for efficient packing of the C array.

Table B-2 HITS Subroutine List (Concl'd)

SUBROUTINE	FUNCTION
TG1795	Dummy alternate trajectory module preset routine.
TG2440	Dummy alternate trajectory module preset routine.
TG2987	Initializes trajectory parameters at preset (default) values.
TRAJT	Computes Real World projectile velocity and range as a function of time, including the influence of oscillatory motion on drag.
TRAJX	Computes Real World projectile velocity and time as a function of range, including the influence of oscillatory motion on drag.
TRAJXF	Computes Fire Control projectile velocity and time as a function of range, including the influence of oscillatory motion on drag.
TVX	Computes projectile velocity and range as a function of time for a particle (no oscillatory drag) trajectory.
T4NTV	Computes mean value, variance and tolerance for arbitrarily distributed variables.
VXT	Computes projectile velocity and time as a function of range for a particle (no oscillatory drag) trajectory.
WIND	Computes projectile horizontal velocity and position brought about by crosswinds.
XXC	Computes the range at which the projectile's oscillatory motion has converged.
ZATAN2	Computes the arctangent of a function, placing the resulting angle in the proper quadrant. Includes provisions to avoid computational errors for the special case of non-oscillatory motion.
ZZ	Computes appropriate value of h as defined by Eq. (C.3-44)

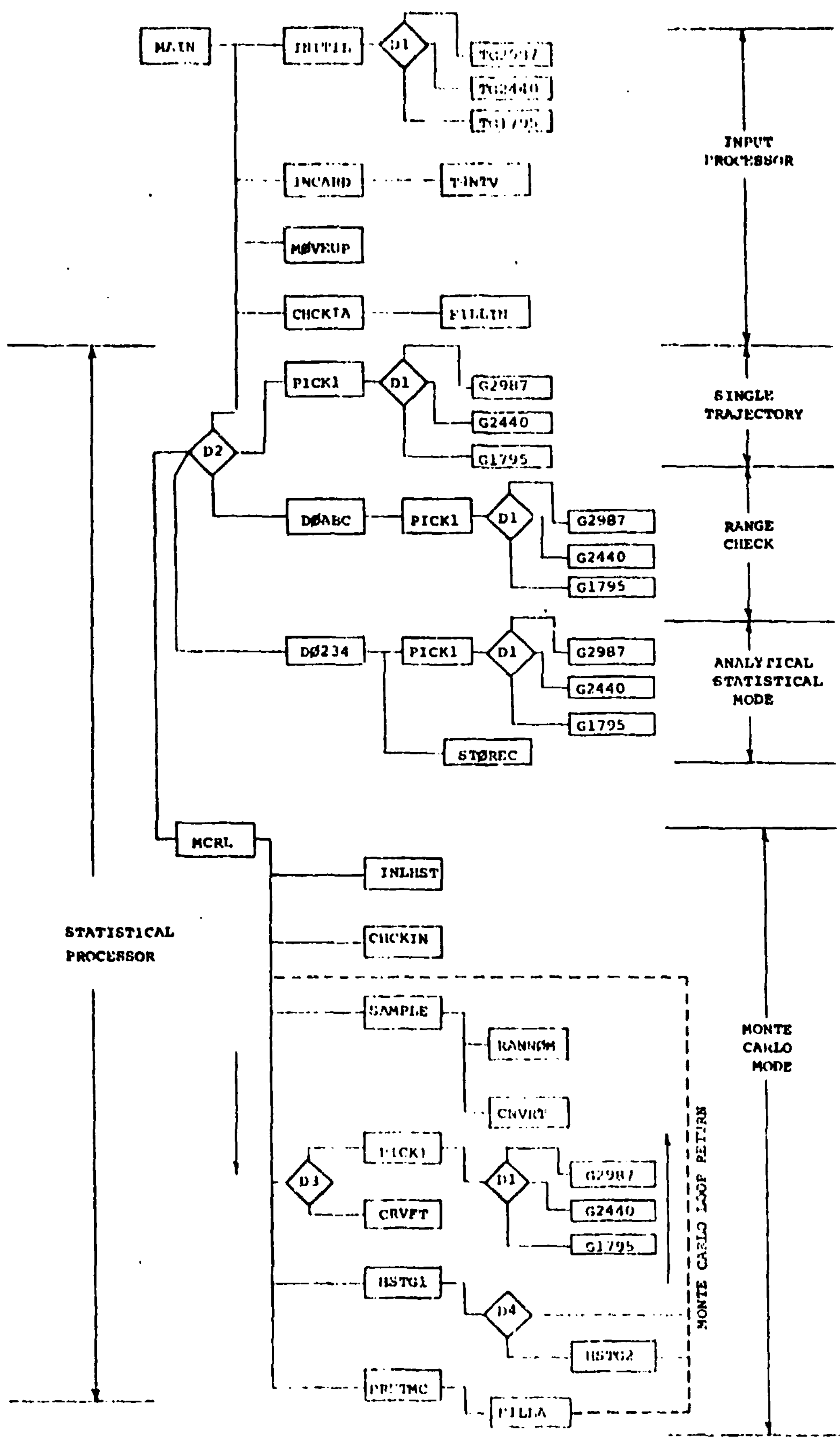


Figure B-4 HITS Subroutine Linking Diagram

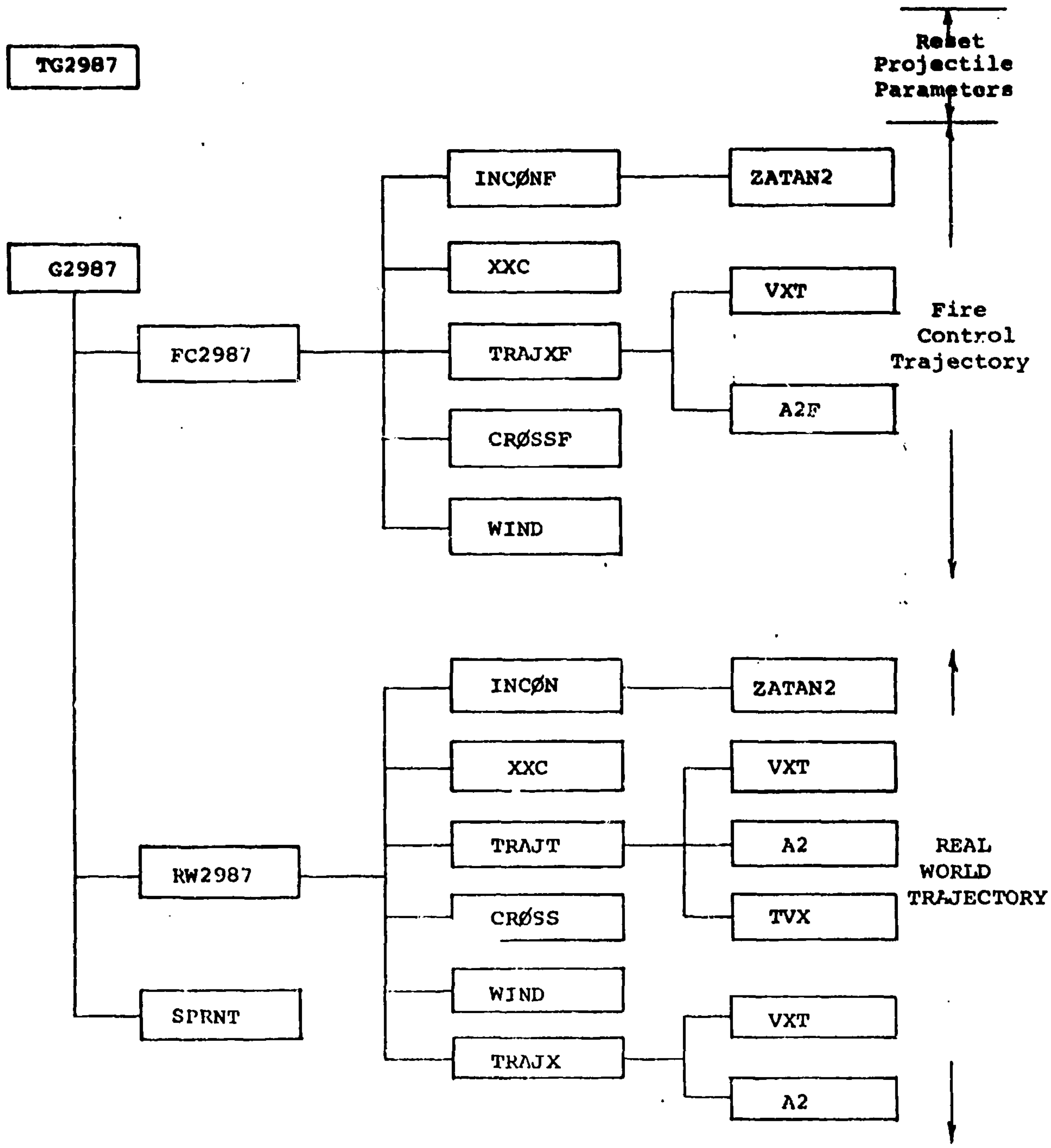


Figure B-5 Projectile Trajectory Module Subroutine Linking Diagram associated with the histogram. These are, at most, KD + K234 variables, where KD is the number of dependent variables (appearing first) and K234 is the number of independent variables (appearing last). The first subscript defines specific items relating to the variable as defined by Figure B-6. The actual histograms are stored in the three-dimensional array ZDATAD (I,J,K). The index K denotes the KD + K234 variables with the dependent variables appearing first and the independent last. Figure B-7 illustrates the format of a two-dimensional slice of the array, ZDATAD(I,J,*). The upper portion applies to the dependent variables and the lower defines the array for independent variables.

	TITLE (I,*)			
ROW I	INTERNAL NAME	DESCRIPTION	REMARKS	
1	IPASS	Histogram Indicator	Variable Ranges \ Dependent, Independent Internal: 1, 2 User: 3, 4	
2	IV	Code Number	X-Axis Label	
3	ITYPE	Variable Type	2, 3, 4, 7 or 8	
4	NRR	Number of Rejects	Rejects	Internal Ranges (IPASS = 1 or 2)
5	CM	Histogram Mean	MEAN	
6	CV	Histogram Variance	VAR	
7	PM	Estimated Mean	(MEAN)	
8	PV	Estimated Variance	(VAR)	
9	FCTR	Histogram Cell Size	Interval	
10	PT	Tolerance	TOL	
11	NRR	Number of Rejects		User Ranges (IPASS = 3 or 4)
12	CM	Histogram Mean	MEAN	
13	CV	Histogram Variance	VAR	
14	PM	Estimated Mean	(MEAN)	
15	PV	Estimated Variance	(VAR)	
16	FCTR	Histogram Range	Interval	
17	PT	Tolerance	TOL	
18	NCELL	Number of Cells		
19	SAVE	Maximum Relative Frequency		
20	ISMØØD	Indicator for Spline Smoothing Option	1 for TYPE = 2, zero	

Figure B-6 Histogram Title Array

I \ J	Internal Ranges X-Axis	Internal Ranges Y-Axis	User Ranges X-Axis	User Ranges Y-Axis
	1	2	3	4
1	X_{b_o}	$100\ N_o/N_T^*$	X_{b_o}	$100\ N_o/N_T^*$
2	X_{b_1}	$100\ N_1/N_T^*$	X_{b_1}	$100\ N_1/N_T^*$

(a) Dependent Variables

I \ J	Internal Ranges X-Axis	Internal Ranges Y-Axis	User Ranges Y-Axis	Theoretical Y-Axis
	1	2	3	4
1				
2				

(b) Independent Variables

Figure B-7 Histogram Storage Array ZDATAD (I,J,*)

APPENDIX C

CLOSED FORM TRAJECTORY EQUATIONS

This appendix derives closed form expressions for the trajectory of a hypervelocity projectile. The equations define the trajectory in time as well as space. The closed form expressions result from realistic simplifying approximations to the full six degree of freedom (6 DOF) equations of motion. All assumptions were verified by comparison to exact solutions as determined by numerical integration of the equations of motion. In all cases, the closed form equations agreed to within engineering tolerances. The equations are of general interest. Avco has employed them in: (1) the aerodynamic design of a hypervelocity projectile, (2) the interpretation of ballistic range test data, and (3) the evaluation of projectile dispersion. Potential applications include incorporation in operational fire control computers, computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination. Section C.1 presents background material and an overview of the analytical development. Particle trajectory equations are presented in Section C.2. Sections C.3 and C.4 present crossrange and downrange perturbation equations, respectively, used to correct the basic particle trajectory.

C.1 Introduction

The complete coupled, nonlinear equations of motion of a projectile in free flight cannot be solved in a general closed form. To obtain such solutions, it is necessary to make simplifying approximations. Many such solutions have been developed in the past under various assumptions for application to a wide variety of problems. These include the gross evaluation of range-velocity-time histories, ballistic range data reduction, and the development of jump angle expressions for specific launch disturbances. One of the unique aspects of the trajectory equations presented here is that both downrange as well as crossrange dynamics are taken into account. The key has been to build upon the existing work and to piece together various solutions in such a manner as to accurately describe the downrange and crossrange dynamics of hypervelocity projectiles.

The trajectory model has three parts. The first part describes the basic particle trajectory including the effects of constant velocity winds in both the crossrange and downrange directions. The second part deals with angle of attack oscillatory motion and subsequent crossrange deflection from the particle trajectory. The final part of the model makes use of the angle of attack history from the oscillatory motion solution to compute range, velocity, and flight time perturbations for correction of the basic particle trajectory. These values account for the increase in drag brought about by angle of attack oscillations. The three parts are then superimposed to describe the complete trajectory history in six degrees of freedom.

The coordinate system used in the trajectory model is shown in Figure C-1. A consistent set of symbols is used throughout this appendix and is presented in Table C-1 along with code numbers which relate the variables to the HITS Code.

C.2 Particle Trajectory

This section presents the basic particle trajectory closed form equations. The development of the equations is discussed in Section C.2.1. Section C.2.2 presents results which verify the equations.

C.2.1 Analytical Development

The particle trajectory solution is fundamentally the same as that developed and reported in an earlier study,[1] with the formulation extended to include the effects of winds in the crossrange and downrange directions. The primary assumptions involved in the solution are the following:

- The drag coefficient varies as $1/V^2$.
- The wind has constant speed and direction.
- The projectile weathercocks into the wind and flies a static zero angle of attack trajectory.
- A uniform density atmosphere.
- The absence of gravity.

The drag coefficient is allowed to vary with velocity according to

$$C_D = C_{D_\infty} + \frac{K_D}{V^2} \qquad \text{(C.2-1)}$$

---

[1]Avco Systems Division, "A Study of the Feasibility of the Two Stage Light Gas Gun as an Air Defense Weapon," AVSD-0197-73-CR, June 1973.

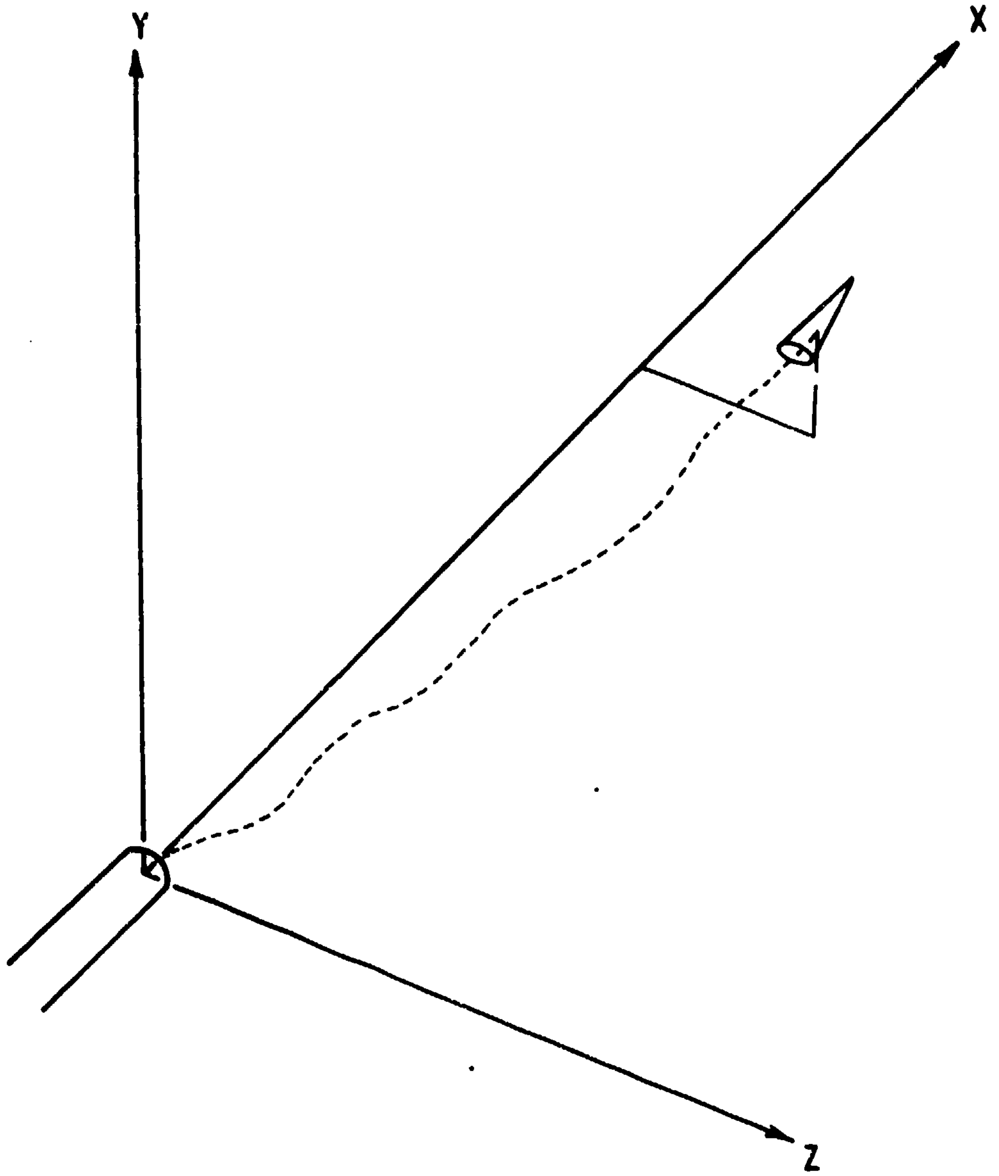


FIGURE C-1 TRAJECTORY COORDINATE SYSTEM

TABLE C-1

LIST OF SYMBOLS

HITS CODE NUMBERS R.W.[1]	F.C.[2]	C.Q.[3]	SYMBOL	DEFINITION AND COMMENTS
122	22		A	reference area, ft^2
		404	B	$1 + \left(\frac{W_Z}{V_o - W_X}\right)^2$
			C_D	drag coefficient
		412	C_{D_∞}	asymptotic limit of C_D as $V \rightarrow \infty$
		419	$\overline{C_{D_\alpha}}$	mean drag coefficient including oscillatory motion effects
		418	$\overline{C_{D_{\alpha=0}}}$	mean drag coefficient excluding oscillatory motion effects
			C_{L_α}	$C_{N_\alpha} - \left(C_{D_\infty} + \frac{K_D}{V_o^2}\right)$
			C_M	pitching moment coefficient about center of gravity, $\frac{M}{\frac{1}{2}\rho V^2 AD}$
		413	C_{M_α}	pitching moment coefficient slope, $\frac{dC_M}{d\alpha}$, rad^{-1}
113	13		$C_{M_{p\alpha}}$	magnus moment coefficient, $\frac{\partial^2 C_M}{\partial\left(\frac{PD}{2V}\right)\partial\alpha}$, rad^{-1}
		415	$C_{M_{\dot\phi\alpha}}$	$\frac{\partial^2 C_M}{\partial P \partial \alpha}$, sec
112	12		C_{M_q}	pitch damping derivative, $\frac{\partial C_M}{\partial\left(\frac{\theta D}{2V}\right)}$

1 Denotes Real World projectile parameter
2 Denotes Fire Control projectile parameter
3 Denotes Computed Quantity

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS R.W.	F.C.	C.Q.	SYMBOL	DEFINITION AND COMMENTS
		414	$C_{M_{\dot\theta}}$	$\frac{\partial C_M}{\partial \dot\theta}$, sec
			C_N	normal force coefficient, $\frac{F}{\frac{1}{2}\rho V^2 A}$
110	10		C_{N_α}	normal force coefficient slope, $\frac{\partial C_N}{\partial \alpha}$, rad^{-1}
			C_x	axial force coefficient, $\frac{F}{\frac{1}{2}\rho V^2 A}$
			C_{x_o}	axial force coefficient at zero angle of attack
124	24		D	reference length, ft
		434	f	$\frac{\rho A_g}{2W}$, ft^{-1}
			g	gravitational acceleration, 32.174 ft/sec^2
		435	h	$\frac{C_{N_\alpha} V_o}{m'}$ (ICNCL = 0); $\frac{C_{L_\alpha} V_O}{m'}$ (ICNCL = 1)
206	206		ICNCL	Aerodynamic coordinate frame selector
126	26		I_x	roll moment of inertia, slug-ft^2
137	27		I_y	pitch moment of inertia, slug-ft^2
		430	I'	$\frac{I_y}{\frac{1}{2}\rho V^2 AD}$, sec^2
		473	JA	jump angle, radians

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS R.W.	F.C.	C.O.	SYMBOL	DEFINITION AND COMMENTS
		471	JA_y	vertical component of jump angle, radians
		472	JA_z	horizontal component of jump angle, radians
130	30		K_{C_X}	constant describing variation of C_X with angle of attack, rad^{-2}
		411	K_D	constant describing variation of C_X with velocity, sec^2/ft^2
		416,417	K_1, K_2	parameters defined by Eqs. (C.3-14) and (C.3-15)
123	23		L	projectile length, ft
		431	m'	$\frac{2W}{\rho A_g V_o}$, sec
128	28		P	projectile spin rate, rad/sec
			P_{cr}	projectile resonant spin rate, rad/sec
		441,442, 443,444	R_1, R_2 R_3, R_4	parameters defined by Eqs. (C.3-16) through (C.3-19)
		445	R_{trim}	static trim angle amplification factor due to spin
111	11		SM	projectile static margin, $\frac{X_{cp} - X_{cg}}{L}$
			t	flight time, sec
		450	t'	$\frac{x}{V_o}$, sec
		500	t_n	nominal flight time, sec

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS R.W.	F.C.	C.Q.	SYMBOL	DEFINITION AND COMMENTS
			Δt	perturbation to flight time arising from oscillatory motion, sec
			V	velocity, ft/sec
			V_x	downrange component of velocity, ft/sec
			V_y	vertical component of velocity, ft/sec
			V_z	horizontal component of velocity, ft/sec
			V_{z_w}	projectile horizontal velocity due to crosswind, ft/sec
101	1		V_o	muzzle velocity, ft/sec
			ΔV	velocity perturbation due to oscillatory motion, ft/sec
125	25		W	projectile weight, lbs
102	2		W_x	downrange component of wind velocity, ft/sec
103	3		W_z	crossrange component of wind velocity, ft/sec
			X	range, ft
			X_c	range at angle of attack convergence, ft
			X_{cg}	projectile center of gravity location from nose, ft
			X_{cp}	projectile center of pressure location from nose, ft
			ΔX	downrange error, ft
			Y	vertical displacement, ft

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS R.W	F.C.	C.Q.	SYMBOL	DEFINITION AND COMMENTS
		426	$\dot{Y}_o$	component of initial velocity in the Y direction, ft/sec
			Z	horizontal displacement, ft
		427	$\dot{Z}_o$	component of initial velocity in the Z direction, ft/sec
			Z_W	projectile horizontal displacement due to cross-wind, ft
		400	α	pitch angle of attack in body fixed coordinate system, radians
114	14		α_{ST}	static trim pitch angle of attack (zero spin rate), radians
		402	α_{trim}	trim pitch angle of attack when spinning, radians
		476	α_o	initial pitch angle of attack, radians
		403	$\dot{\alpha}_o$	time rate of change of α_o , rad/sec
		401	$\bar{\alpha}$	total angle of attack, radians
		409	$\bar{\alpha}_{upper}$	upper envelope of $\bar{\alpha}$, radians
		408	β	yaw angle of attack in body, fixed coordinate system, radians
115	15		β_{ST}	static trim yaw angle of attack (zero spin rate), radians
		405	β_{trim}	trim yaw angle of attack when spinning, radians
		406	β_o	initial yaw angle of attack, radians
		407	$\dot{\beta}_o$	time rate of change of β_o , rad/sec

TABLE C-1

LIST OF SYMBOLS (Concl'd)

HITS CODE NUMBERS R.W.	F.C.	C.Q.	SYMBOL	DEFINITION AND COMMENTS
116	16		θ_o	initial pitch orientation, radians
118	18		$\dot{\theta}_o$	initial pitching rate, rad/sec
		464,465, 466,429	$\lambda_0, \lambda_1, \lambda_2, \Delta\lambda$	components of damping rates, sec^{-1}
		467,463	ν_1, ν_2	phase angles defined by Eqs. (C.3-24) and (C.3-25), respectively, radians
104	4		ρ	atmospheric density, $slugs/ft^3$
		437,438, 439	ϕ_0, ϕ_1, ϕ_2	phase angles defined by Eqs. (C.4-15), (C.3-45), and (C.3-46), respectively, radians
			ϕ_{ST}	meridianal orientation of static trim, radians
			$\Delta\phi$	meridianal shift of static trim orientation due to spin, radians
117	17		ψ_o	initial yaw orientation, radians
119	19		$\dot{\psi}_o$	initial yaw rate, rad/sec
		460,461, 462,428	$\omega_0, \omega_1, \omega_2, \Delta\omega$	components of oscillation frequency defined by Eqs. (C.3-7), (C.3-3), (C.3-4), and (C.3-8), respectively; rad/sec

where C_{D_∞} and K_D are constants evaluated for the specific projectile and velocity regime. Knowledge of the projectile drag coefficient at two velocities within the range of interest allows determination of the two constants C_{D_∞} and K_D. The drag coefficient approximation may also be expressed in terms of the ballistic coefficient, β (i.e., $\beta = W/C_DA$), as indicated in Figure C-2 which shows the accuracy of a typical fit.

With the inclusion of constant velocity winds, the previous[1] particle trajectory solution takes the form

$$V_x = \left\{ (V_o - W_x)^2 e^{2f\sqrt{B}\,C_{D_\infty}(W_x t - x)} + \frac{K_D}{B C_{D_\infty}} \left(e^{2f\sqrt{B}\,C_{D_\infty}(W_x t - x)} - 1 \right) \right\}^{1/2} + W_x \qquad \text{(C.2-2)}$$

$$t = \frac{1}{f\sqrt{C_{D_\infty} K_D}} \left\{ \tan^{-1}\left[\sqrt{\frac{B C_{D_\infty}}{K_D}}\,(V_o - W_x) \right] - \tan^{-1}\left[\sqrt{\frac{B C_{D_\infty}}{K_D}}\,(V_x - W_x) \right] \right\} \qquad \text{(C.2-3)}$$

$$V_{z_w} = W_z \left[1 - \frac{V_x - W_x}{V_o - W_x} \right] \qquad \text{(C.2-4)}$$

$$Z_w = \frac{W_z}{V_o - W_x}(V_o t - x) \qquad \text{(C.2-5)}$$

It is most important to notice that range is the independent variable in the particle trajectory equations. Subsequently, velocity and time-of-flight corrections are computed and applied as a function of range and not time. Equation (C.2-3) is indeterminate for the case of constant drag coefficient ($C_D = C_{D_\infty}$ and $K_D = 0$). Numerical evaluation of this case can be made using any sufficiently small value for K_D, such that $K_D/V^2 << C_{D_\infty}$. In the presence of a wind component in the downrange direction, W_x, an iterative procedure must be used to solve the equations, since the product of the unknown flight time, t, and W_x appears in the expression for velocity, i.e., Eq. (C.2-2).

---

[1]Avco Systems Division, "A Study of the Feasibility of the Two Stage Light Gas Gun as an Air Defense Weapon," AVSD-0197-73-CR, June 1973.

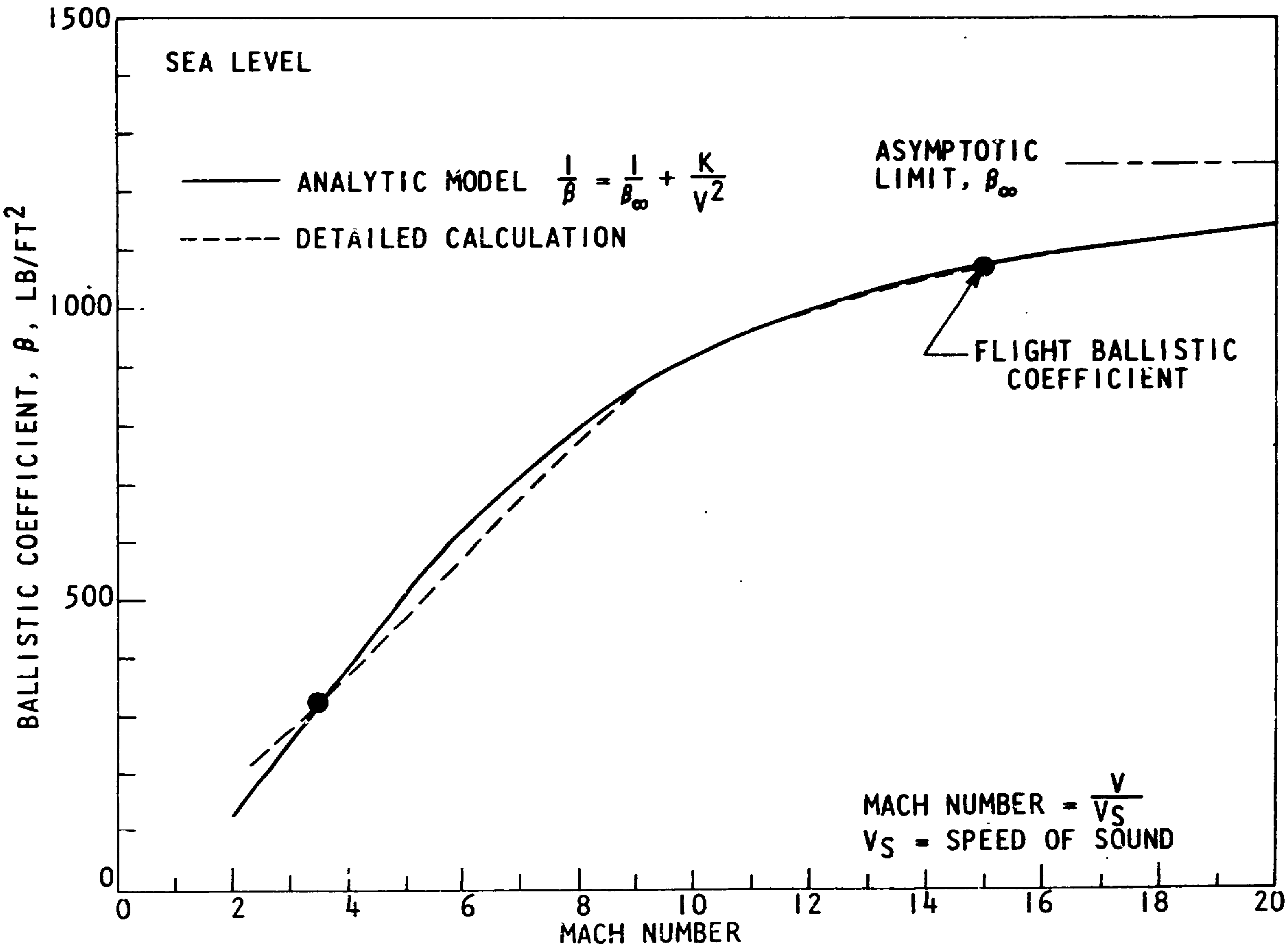


FIGURE C-2 TYPICAL BALLISTIC COEFFICIENT VARIATION WITH MACH NUMBER

C.2.2 Verification

The particle trajectory equations, including the drag coefficient approximation, were verified by comparing them to exact numerical solutions of the 6 DOF equations of motion. Figure C-3 summarizes one comparison. It shows the range-velocity-time histories from the analytic model and the 6 DOF simulation agree.

A separate comparison was made to evaluate the effects of winds on crossrange error. This comparison (as well as all subsequent check cases in this appendix) was made using a smaller projectile whose aerodynamic and mass properties are listed in Table C-2. With a muzzle velocity of 11,000 ft/sec and a constant crossrange wind velocity, W_z, of 15 ft/sec, the 6 DOF calculations showed an 11.2 ft crossrange drift after 2 seconds of flight. The analytic model agreed to within 2 percent.

C.3 Crossrange Perturbations

This section presents the crossrange perturbation equations that are used to correct the basic particle trajectory. The development of the equations is discussed in Section C.3.1. Section C.3.2 presents results which verify the equations.

C.3.1 Analytical Development

The crossrange perturbation model addresses the oscillatory motion and corresponding crossrange effects. A solution for the angle of attack history was obtained from a NACA report.[1] This model employs the following typical ballistic range assumptions:

- Constant velocity
- Constant spin rate
- Linear aerodynamics
- Small trim angles due to shape or mass asymmetries
- Small angles ($\cos \Theta \approx 1$, $\tan \Theta \approx \sin \Theta \approx \Theta$)
- Uniform density atmosphere
- The absence gravity

---

[1]Nelson, Robert L., "The Motion of Rolling Symmetrical Missiles Referred to a Body-Axis System," NACA Technical Note 3737, November 1956.

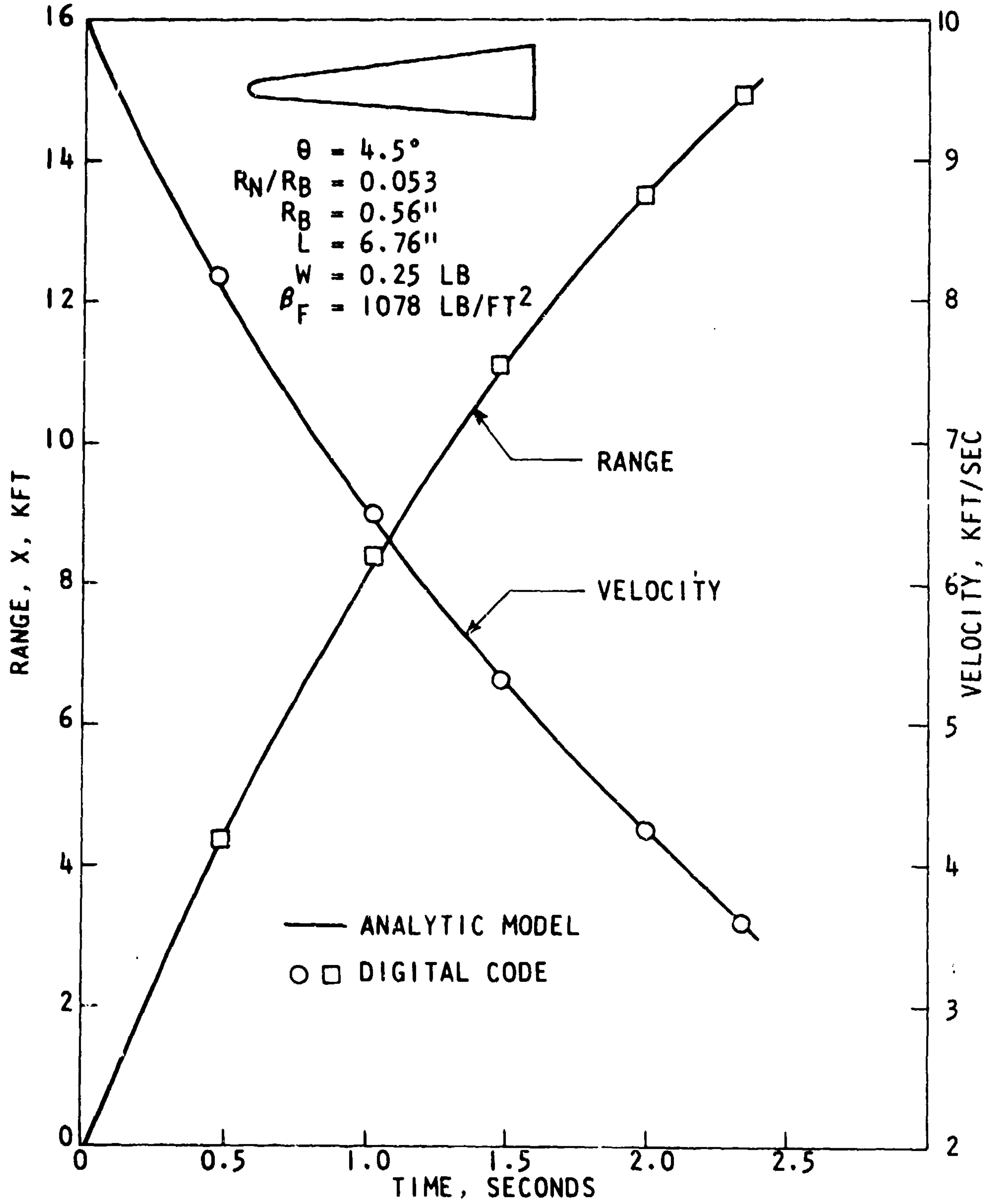


FIGURE C-3 TRAJECTORY COMPARISON

TABLE C-2

TEST CASE PROJECTILE PROPERTIES

CHARACTERISTIC	VALUE
Mass Properties:	
Cone Angle, Θ_C	5.5°
Length, L	3.7"
Nose Radius, R_N	.019"
Base Radius, R_B	.375"
Weight, W	.11 lbs.
Roll Moment of Inertia, I_X	$.538 \times 10^{-6}$ slug-ft^2
Pitch Moment of Inertia, I_Y	$.787 \times 10^{-5}$ slug-ft^2
Aerodynamic Characteristics	
Flight Ballistic Coefficient (M = 15), β_F	1000 PSF
Static Margin, SM	6.2% L

At first appearance, the constant velocity assumption may seem objectionable since significant velocity variations occur along the actual trajectory. The constant velocity assumption is required to achieve a closed form solution with a nonzero spin rate. In reality, the method of applying the constant velocity crossrange corrections to the particle trajectory minimizes the errors incurred in calculating the perturbation assuming a constant velocity. Velocity affects crossrange displacement through its influence on the aerodynamic force in the crossrange direction. This force is composed of two components: the drag force and the normal force. For low drag, slender projectiles, the normal force dominates and the drag force can be safely ignored. The normal force is proportional to the angle of attack. Thus, assuming small trim angles of attack, the normal force is only important while the projectile is oscillating in angle of attack. Furthermore, the first few oscillations are the most important because subsequent oscillations tend to average out. Since these occur near the muzzle when velocity is essentially constant, the constant velocity assumption is realistic. The crossrange perturbation is applied to the basic particle trajectory as a function of range. This scales the "constant velocity" crossrange perturbation in accord with the downrange velocity variation of the particle trajectory. Thus, the crossrange perturbation equations simulate the true velocity variation.

The solution given in the NACA report[1] is in terms of the body fixed coordinate system shown in Figure C-4. The form of the solution is

$$\alpha = K_1 e^{\lambda_1 t'} \sin(\omega_1 t' + \nu_1) - K_2 e^{\lambda_2 t'} \sin(\omega_2 t' - \nu_2) + \alpha_{trim} \qquad (C.3\text{-}1)$$

$$\beta = K_1 e^{\lambda_1 t'} \cos(\omega_1 t' + \nu_1) + K_2 e^{\lambda_2 t'} \cos(\omega_2 t' - \nu_2) + \beta_{trim} \qquad (C.3\text{-}2)$$

where α and β are, respectively, the pitch and yaw body angles of attack as illustrated in Figure C-4. These equations are written in terms of the range expressed as a pseudo flight time, $t' = X/V_O$. This effects the appropriate scaling between

---

[1]Nelson, Robert L., "The Motion of Rolling Symmetrical Missiles Referred to a Body-Axis System," NACA Technical Note 3737, November 1956.

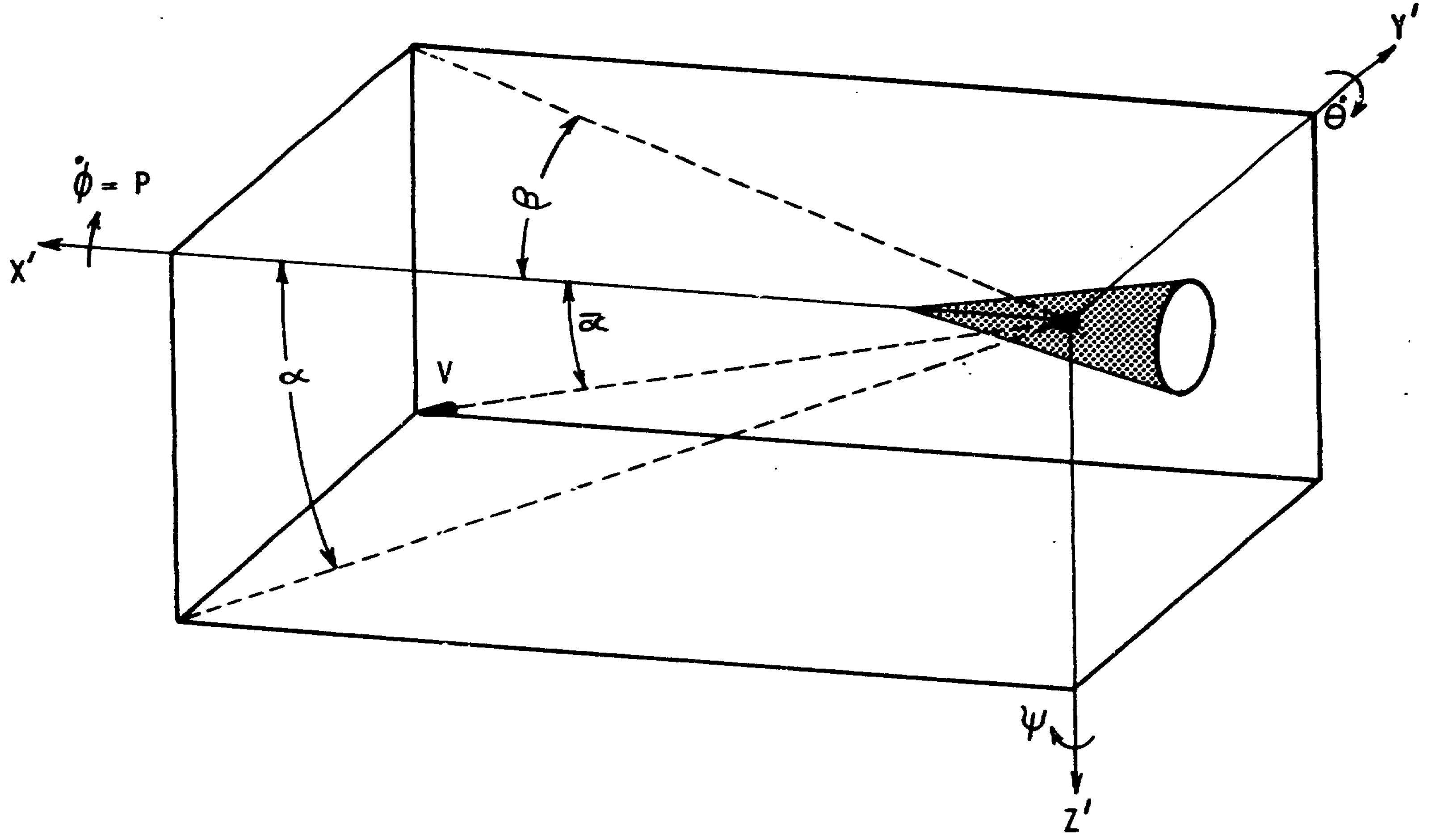


FIGURE C-4 PROJECTILE BODY FIXED COORDINATE SYSTEM the crossrange perturbation solution assuming constant velocity and the realistic velocity variations contained in the particle trajectory. The frequency and damping constants are defined as follows:

$$\omega_1 = \omega_o - \Delta\omega \tag{C.3-3}$$

$$\omega_2 = \omega_o + \Delta\omega \tag{C.3-4}$$

$$\lambda_1 = \lambda_o + \Delta\lambda \tag{C.3-5}$$

$$\lambda_2 = \lambda_o - \Delta\lambda \tag{C.3-6}$$

where

$$\omega_o = \frac{\sqrt{2}}{4}\left\{-\frac{4C_{M_\alpha}}{I'} + \left(P\frac{I_x}{I_y}\right)^2 - \left(\frac{C_{N_\alpha}}{m'} + \frac{C_{M_{\dot\theta}}}{I'}\right)^2 + \left[\left\{-\frac{4C_{M_\alpha}}{I'} + \left(P\frac{I_x}{I_y}\right)^2 - \left(\frac{C_{N_\alpha}}{m'} + \frac{C_{M_{\dot\theta}}}{I'}\right)^2\right\}^2 + 4\left[P\frac{I_x}{I_y}\left(\frac{C_{N_\alpha}}{m'} + \frac{C_{M_{\dot\theta}}}{I'}\right) + 2P\frac{C_{M_{\dot\phi_\alpha}}}{I'}\right]^2\right]^{1/2}\right\}^{1/2} \tag{C.3-7}$$

$$\Delta\omega = P\left(1 - \frac{I_x}{2I_y}\right) \tag{C.3-8}$$

$$\lambda_o = \frac{1}{2}\left(\frac{C_{M_{\dot\theta}}}{I'} - \frac{C_{N_\alpha}}{m'}\right) \tag{C.3-9}$$

$$\Delta\lambda = \frac{P I_x}{4\omega_o I_y}\left(\frac{C_{N_\alpha}}{m'} + \frac{C_{M_{\dot\theta}}}{I'}\right) + \frac{P C_{M_{\dot\phi_\alpha}}}{2\omega_o I'} \tag{C.3-10}$$

where

$$C_{M_\alpha} = -C_{N_\alpha}(SM)\frac{L}{D} \tag{C.3-11}$$

$$I' = \frac{2I_y}{\rho A D V_o^2} \tag{C.3-12}$$

$$m' = \frac{2W}{\rho A g V_o} \tag{C.3-13}$$

The terms K_1 and K_2 in Eqs. (C.3-1) and (C.3-2) are constants involving the initial conditions

$$K_1 = \sqrt{R_1^2 + R_2^2} \tag{C.3-14}$$

$$K_2 = \sqrt{R_3^2 + R_4^2} \tag{C.3-15}$$

where

$$R_1 = \frac{\omega_o[\dot{\alpha}_o + \omega_2(\beta_o - \beta_{trim}) - \lambda_2(\alpha_o - \alpha_{trim})] + \Delta\lambda\,[\dot{\beta}_o - \omega_2(\alpha_o - \alpha_{trim}) - \lambda_2(\beta_o - \beta_{trim})]}{2(\omega_o^2 + \Delta\lambda^2)} \tag{C.3-16}$$

$$R_2 = \frac{-\omega_o[\dot{\beta}_o - \omega_2(\alpha_o - \alpha_{trim}) - \lambda_2(\beta_o - \beta_{trim})] + \Delta\lambda\,[\dot{\alpha}_o + \omega_2(\beta_o - \beta_{trim}) - \lambda_2(\alpha_o - \alpha_{trim})]}{2(\omega_o^2 + \Delta\lambda^2)} \tag{C.3-17}$$

$$R_3 = \beta_o - \beta_{trim} - R_1 \tag{C.3-18}$$

$$R_4 = \alpha_o - \alpha_{trim} - R_2 \tag{C.3-19}$$

The initial conditions α_o, β_o, $\dot{\alpha}_o$, and $\dot{\beta}_o$ in Eqs. (C.3-16) and (C.3-17) are related to the projectile inertial orientations and rates by

$$\alpha_o = \theta_o - \frac{\dot{Y}_o}{V_o} \tag{C.3-20}$$

$$\beta_o = \frac{\dot{Z}_o}{V_o} - \psi_o \tag{C.3-21}$$

$$\dot{\alpha}_o = \dot{\theta}_o - \frac{\rho A g}{2w} C_{N_\alpha} V_o \alpha_o - P\beta_o \tag{C.3-22}$$

$$\dot{\beta}_o = -\dot{\psi}_o - \frac{\rho A g}{2w} C_{N_\alpha} V_o \beta_o + P\alpha_o \tag{C.3-23}$$

The phase angles ν_1 and ν_2 in Eqs. (C.3-1) and (C.3-2) are computed according to

$$\nu_1 = \sin^{-1}\left(\frac{R_2}{K_1}\right) = \cos^{-1}\left(\frac{R_1}{K_1}\right) \tag{C.3-24}$$

$$\nu_2 = \sin^{-1}\left(\frac{R_4}{K_2}\right) = \cos^{-1}\left(\frac{R_3}{K_2}\right) \tag{C.3-25}$$

Both the sine and cosine definitions are given here to establish the proper quadrants in which the phase angles ν_1 and ν_2 lie.

The terms α_{trim} and β_{trim} in Eqs. (C.3-1), (C.3-2), (C.3-16), and (C.3-17) are body fixed trim values for a spinning body (i.e., the "rolling trim") and are related to the body fixed static trim values, α_{ST} and β_{ST}, by

$$\alpha_{trim} = R_{trim}\sqrt{\alpha_{ST}^2 + \beta_{ST}^2}\,\cos(\phi_{ST} + \Delta\phi) \tag{C.3-26}$$

$$\beta_{trim} = R_{trim}\sqrt{\alpha_{ST}^2 + \beta_{ST}^2}\,\sin(\phi_{ST} + \Delta\phi) \tag{C.3-27}$$

where

$$\phi_{ST} = \sin^{-1}\frac{\beta_{ST}}{\sqrt{\alpha_{ST}^2 + \beta_{ST}^2}} = \cos^{-1}\frac{\alpha_{ST}}{\sqrt{\alpha_{ST}^2 + \beta_{ST}^2}} \tag{C.3-28}$$

From the work presented in an AIAA journal,[1] R_{trim} and $\Delta\phi$ are given by

$$R_{trim} = \left[\left(1 - \left\{\frac{P}{P_{CR}}\right\}^2\right)^2 + \left(2\mu \frac{P}{P_{CR}}\right)^2\right]^{-1/2} \tag{C.3-29}$$

$$\Delta\phi = \begin{cases} \tan^{-1}\left[\dfrac{2\mu\left(\dfrac{P}{P_{CR}}\right)}{\left[1 - \left(\dfrac{P}{P_{CR}}\right)^2\right]}\right] & \text{for } \left[1 - \left(\dfrac{P}{P_{CR}}\right)^2\right] > 0 \\[2em] \pi + \tan^{-1}\left[\dfrac{2\mu\left(\dfrac{P}{P_{CR}}\right)}{1 - \left(\dfrac{P}{P_{CR}}\right)^2}\right] & \text{for } \left[1 - \left(\dfrac{P}{P_{CR}}\right)^2\right] < 0 \end{cases} \tag{C.3-30}$$

where

$$P_{CR} = \left[\frac{-C_{M_\alpha} \rho V_o^2 AD}{2(I_y - I_x)}\right]^{1/2} \tag{C.3-31}$$

$$\mu = \frac{\dfrac{C_{N_\alpha} \rho A g}{4w} - \dfrac{C_{M_q} D^2 \rho A}{8(I_y - I_x)}}{\left[\dfrac{-C_{M_\alpha} \rho AD}{2(I_y - I_x)}\right]^{1/2}} \tag{C.3-32}$$

---

[1] Pettus, Joseph J., "Persistent Reentry Vehicle Roll Resonance," American Institute of Aeronautics and Astronautics Paper No. 66-49, January 1966.

The term R_{trim} represents the amplification factor for a spin rate, P, close to the critical (or resonant) frequency, P_{CR}. The term $\Delta\phi$ represents the meridianal shift of the plane of the rolling trim depending on the value of the spin rate with respect to P_{CR}. At resonance conditions ($P = P_{CR}$), the amplification is maximum and the rolling trim plane is shifted by 90° with respect to the plane of the static trim.

With the angle of attack history established, the lift force is known. The crossrange displacement is found by integration of the equations of motion normal to the flight path. The remainder of this section is devoted to this integration.

Because of the spinning motion of the projectile, the body fixed coordinate system in which α and β are described, see Figure C-4, rotates with respect to the inertial frame in which the trajectory is described, see Figure C-1. At exit from the muzzle (t = 0) the two coordinate systems are aligned. At this instant, positive α is nose up and creates a lift force in the vertical direction which is the positive y direction in the trajectory frame. Owing to the nature of the body fixed coordinate system, positive β at t = 0 is nose left and produces a horizontal yaw force in the negative z direction. With positive spin being "right wing down," the body fixed lift and yaw forces rotate with respect to the trajectory coordinate frame as illustrated in Figure C-5. The force components in the inertial trajectory coordinate system are written as follows.[1]

$$L_\alpha = \frac{C_{N_\alpha} \rho A V_o^2}{2} \alpha \qquad (C.3\text{-}33)$$

$$L_\beta = \frac{C_{N_\alpha} \rho A V_o^2}{2} \beta \qquad (C.3\text{-}34)$$

$$\phi = P t' \qquad (C.3\text{-}35)$$

$$F_y = L_\alpha \cos\phi + L_\beta \sin\phi \qquad (C.3\text{-}36)$$

$$F_z = L_\alpha \sin\phi - L_\beta \cos\phi \qquad (C.3\text{-}37)$$

1 In most applications, it is more appropriate to use C_{L_α} instead of C_{N_α} in the ensuing development. This affects only the definition of h, as noted in Eq. (C.3-44).

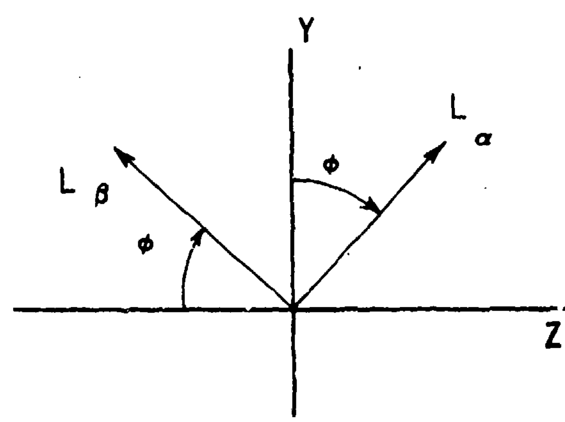


FIGURE C-5 LIFT FORCE COMPONENTS IN THE TRAJECTORY FRAME

Thus the equations of motion are

$$\frac{w}{g}\,\frac{d^2Y}{dt'^2} = \frac{C_{N_\alpha}\,\rho A\,V_o^2}{2}\,[\alpha \cos Pt' + \beta \sin Pt'] \tag{C.3-38}$$

$$\frac{w}{g}\,\frac{d^2Z}{dt'^2} = \frac{C_{N_\alpha}\,\rho A\,V_o^2}{2}\,[\alpha \sin Pt' - \beta \cos Pt'] \tag{C.3-39}$$

where α and β are given by Eqs. (C.3-1) and (C.3-2). The notation t' is maintained in Eqs. (C.3-38) and (C.3-39) since it is not the true flight time.

Integrating Eqs. (C.3-38) and (C.3-39) gives the solutions for crossrange velocity and displacement brought about by oscillatory motion

$$\begin{aligned} V_y = h\frac{V}{V_o}\Bigg\{ & \frac{K_1}{\sqrt{\lambda_i^2 + (\omega_1 + P)^2}}\left[e^{\lambda_1 t'} \sin\{(\omega_1 + P)t' + \nu_1 + \phi_1\} - \sin(\nu_1 + \phi_1)\right] \\ & - \frac{K_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}}\left[e^{\lambda_2 t'} \sin\{(\omega_2 - P)t' - \nu_2 + \phi_2\} - \sin(\phi_2 - \nu_2)\right] \\ & + \frac{\alpha_{trim}}{P}\sin Pt' - \frac{\beta_{trim}}{P}(\cos Pt' - 1) + \frac{1}{h}\left(\frac{dy}{dt}\right)_o\Bigg\} \end{aligned} \tag{C.3-40}$$

$$V_z = h \frac{V}{V_o} \left\{ \frac{-K_1}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} \left[e^{\lambda_1 t'} \cos |(\omega_1 + P) t' + \nu_1 + \phi_1| - \cos (\nu_1 + \phi_1) \right] \right.$$

$$- \frac{K_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} \left[e^{\lambda_2 t'} \cos |(\omega_2 - P) t' - \nu_2 + \phi_2| - \cos (\phi_2 - \nu_2) \right]$$

$$\left. - \frac{\beta_{trim}}{P} \sin P t' - \frac{\alpha_{trim}}{P} (\cos P t' - 1) + \frac{1}{h} \left. \frac{dz}{dt} \right)_o \right\} \tag{C.3-41}$$

$$Y = h \left\{ \frac{K_1}{\lambda_1^2 + (\omega_1 + P)^2} e^{\lambda_1 t'} \sin [(\omega_1 + P) t' + \nu_1 + 2\phi_1] \right.$$

$$- \frac{K_2}{\lambda_2^2 + (\omega_2 - P)^2} e^{\lambda_2 t'} \sin [(\omega_2 - P) t' - \nu_2 + 2\phi_2]$$

$$- \left[\frac{K_1 \sin (\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} - \frac{K_2 \sin (\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} - \frac{1}{h} \left. \frac{dy}{dt} \right)_o \right] t'$$

$$- \frac{K_1 \sin (\nu_1 + 2\phi_1)}{\lambda_1^2 + (\omega_1 + P)^2} + \frac{K_2 \sin (2\phi_2 - \nu_2)}{\lambda_2^2 + (\omega_2 - P)^2}$$

$$\left. + \alpha_{trim} \left[\frac{1 - \cos P t'}{P^2} \right] + \beta_{trim} \left[\frac{P t' - \sin P t'}{P^2} \right] \right\} \tag{C.3-42}$$

$$Z = h \left\{ \frac{-K_1}{\lambda_1^2 + (\omega_1 + P)^2} e^{\lambda_1 t'} \cos[(\omega_1 + P)t' + \nu_1 + 2\phi_1] \right.$$

$$- \frac{K_2}{\lambda_2^2 + (\omega_2 - P)^2} e^{\lambda_2 t'} \cos[(\omega_2 - P)t' - \nu_2 + 2\phi_2]$$

$$+ \left[\frac{K_1 \cos(\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} + \frac{K_2 \cos(\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} + \left.\frac{1}{h}\frac{dz}{dt}\right)_o \right] t'$$

$$+ \frac{K_1 \cos(\nu_1 + 2\phi_1)}{\lambda_1^2 + (\omega_1 + P)^2} + \frac{K_2 \cos(2\phi_2 - \nu_2)}{\lambda_2^2 + (\omega_2 - P)^2}$$

$$\left. - \beta_{trim} \left[\frac{1 - \cos P t'}{P^2} \right] + \alpha_{trim} \left[\frac{P t' - \sin P t'}{P^2} \right] \right\} \tag{C.3-43}$$

where [1]

$$h = \begin{cases} \dfrac{C_{N_\alpha} \rho A g V_o^2}{2W} & \text{ICNCL} = 0 \\ \dfrac{C_{L_\alpha} \rho A g V_o^2}{2W} & \text{ICNCL} = 1 \end{cases} \tag{C.3-44}$$

$$\phi_1 = \sin^{-1} \frac{-(\omega_1 + P)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} = \cos^{-1} \frac{\lambda_1}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} \tag{C.3-45}$$

$$\phi_2 = \sin^{-1} \frac{-(\omega_2 - P)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} = \cos^{-1} \frac{\lambda_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} \tag{C.3-46}$$

1. All numerical verification results presented in this appendix use C_{N_α} . This is a worst case assumption for the reference projectile.

again take note of the respective quadrants for ϕ_1 and ϕ_2.

The "jump angle" is defined as the root sum of the squares of the coefficients of the linear term in Eqs. (C.3-42) and (C.3-43) divided by the velocity. It is

$$JA = \sqrt{JA_y^2 + JA_z^2} \tag{C.3-47}$$

where

$$JA_y = \frac{h}{V_o}\left[\frac{K_2 \sin(\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} - \frac{K_1 \sin(\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}}\right] \tag{C.3-48}$$

$$JA_z = \frac{h}{V_o}\left[\frac{K_1 \cos(\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} + \frac{K_2 \cos(\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}}\right] \tag{C.3-49}$$

C.3.2 Verification

Verification of the crossrange perturbation equations was established by comparison to four 6 DOF simulations. The results are summarized in Table C-3. In all cases, the analytic solution agrees to within 6%. An additional comparison is shown in Figure C-6, in which crossrange error due to trim angle of attack is shown as a function of spin rate. The analytic model is seen to be in excellent agreement with the 6 DOF calculations.

C.4 Downrange Perturbations

This section presents the downrange perturbation equations that are used to correct the basic particle trajectory. The development of the equations is discussed in Section C.4.1. Section C.4.2 discusses results which verify the equations.

C.4.1 Analytical Development

The downrange perturbation equations model the downrange effects of angle of attack oscillations. In addition to the assumptions stated in Sectior C.3.1, the following assumptions are required to achieve an accurate closed form solution:

TABLE C-3

TRAJECTORY MODEL VERIFICATION

V_o = 11000 ft/sec

Flight Time = 0.2 Sec

CASE	θ_o	o	θ_o (RAD/SEC)	o (RAD/SEC)	t	t	P (RAD/SEC)		X (FT)	Y (FT)	Z (FT)	V (FT/SEC)	JUMP ANGLE (MILS)
1	0	0	0	0	0	0	0	6 DOF	2091.4	0.0	0.0	9937.0	0.0
								MODEL	2091.4	0.0	0.0	9937.0	0.0
2	10°	0	0	0	0	0	400	6 DOF	2084.4	.37	-.13	9903.5	.19
								MODEL	2085.2	.39	-.13	9902.4	.20
3	0	0	100	0	0	0	400	6 DOF	2090.1	2.75	0	9931.8	1.32
								MODEL	2090.1	2.81	0	9930.5	1.35
4	2	3	-50	70	0.2	0	200	6 DOF	2089.7	-1.29	2.59	9931.8	1.39
								MODEL	2089.8	-1.35	2.68	9928.5	1.44

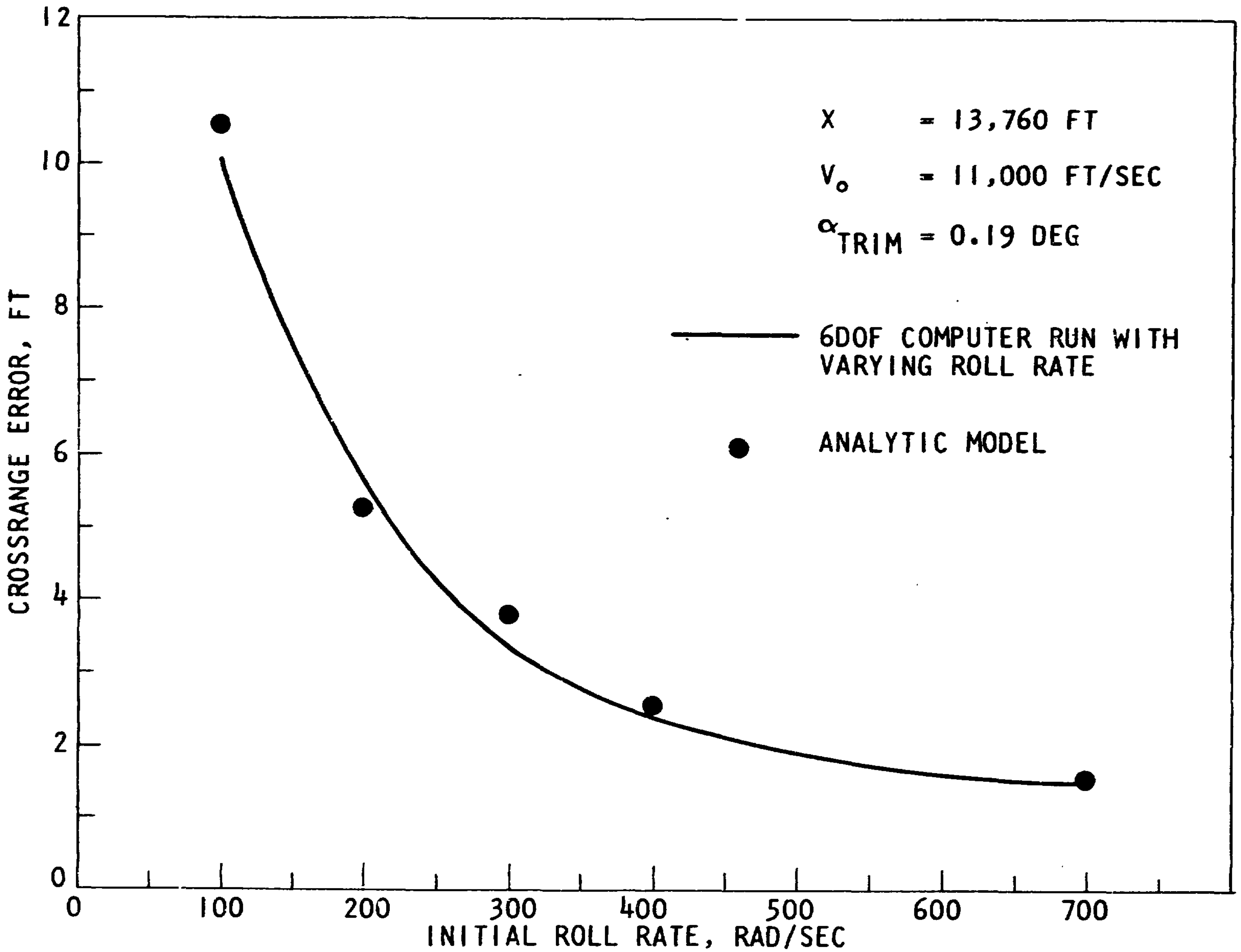


FIGURE C-6 LIFT INDUCED DISPERSION AS A FUNCTION OF ROLL RATE

- Trim angles of attack cause negligible drag.
- Downrange effects can be captured by mean drag coefficients.
- Pitch oscillations are lightly damped.

In addition to the crossrange perturbation, oscillatory motion has a downrange effect. Angle of attack oscillations cause the projectile to experience a net increase in drag. The exact mechanism is illustrated in Figure C-7. As shown in Figure C-7(a), drag is defined as the force acting in line with the velocity vector. There are two contributions when the body is at angle of attack, a component of the normal force, C_N, acts in the drag direction, increasing the drag force. The second component is the axial force coefficient, C_x, which in general is a function of angle of attack, as illustrated in Figure C-7(b). The drag coefficient is

$$C_D = C_x \cos\alpha + C_N \sin\alpha \tag{C.4-1}$$

which, for small angles, becomes

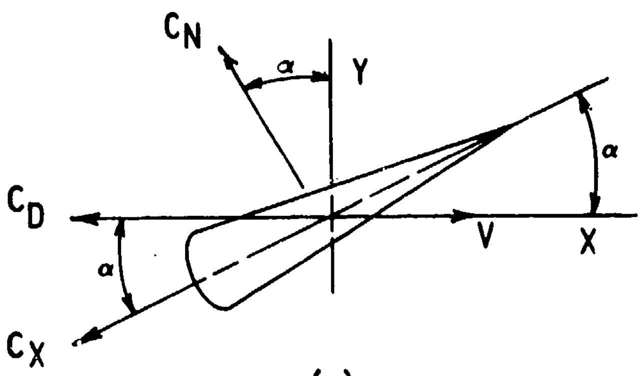
$$C_D \cong C_x + C_N \alpha \tag{C.4-2}$$

Modeling the axial force as parabolic in angle of attack, $C_x = C_{x_o} + K_{C_x}\alpha^2$, and the normal force as linear in angle of attack (i.e., $C_N = C_{N_\alpha}\alpha$) the drag coefficient becomes

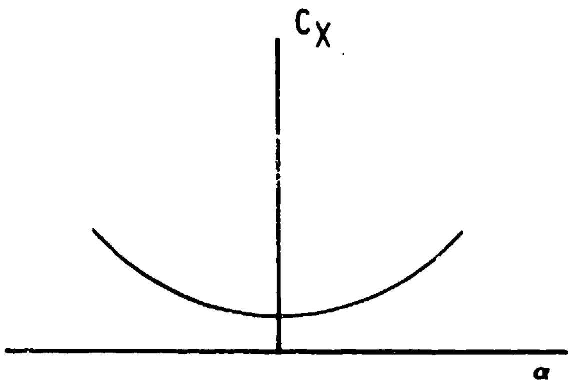
$$C_D = C_{x_o} + (C_{N_\alpha} + K_{C_x})\alpha^2 \tag{C.4-3}$$

The term C_{x_o} is the value of the drag coefficient at zero angle of attack. It is given by the drag model of Section C.2.1, Eq. (C.2-1). For the general case in which oscillations occur in both the pitch and yaw planes, the angle of attack in these expressions becomes the total angle of attack (i.e., the angle between the body axis and the velocity vector), denoted hereafter as $\bar{\alpha}$. For small angles

$$\bar{\alpha}^2 = \alpha^2 + \beta^2 \tag{C.4-4}$$



(a)



(b)

FIGURE C-7 DOWNRANGE FORCES DUE TO ANGLE OF ATTACK where α and β are, respectively, the pitch and yaw angles of attack in the body fixed coordinate system. Upon substitution of Eqs. (C.3-1) and (C.3-2) into Eq. (C.4-4), the total attack (with no trim) becomes

$$\bar{\alpha}^2 = K_1^2 \, e^{\frac{2\lambda_1 x}{V_o}} + K_2^2 \, e^{\frac{2\lambda_2 x}{V_o}} + 2 K_1 K_2 \, e^{\frac{2\lambda_o x}{V_o}} \cos\left(\frac{2\omega_o x}{V_o} + \nu_1 - \nu_2\right) \quad \text{(C.4-5)}$$

where t' has been replaced by X/V_o. The trim angle of attack is neglected so $\bar{\alpha}^2$ may be expressed in a concise, workable form. This is a reasonable approximation since in reality the trim angles are small and do not have an appreciable effect on the drag.

Substitution of Eq. (C.4-5) into Eq. (C.4-3), yields the drag coefficient

$$C_D = C_{D_\infty} + \frac{K_D}{V^2} + (C_{N_\alpha} + K_{C_x}) \left[K_1^2 \, e^{\frac{2\lambda_1 x}{V_o}} + K_2^2 \, e^{\frac{2\lambda_2 x}{V_o}} + 2K_1 K_2 \, e^{\frac{2\lambda_o x}{V_o}} \cos\left(\frac{2\omega_o x}{V_o} + \nu_1 - \nu_2\right)\right] \quad \text{(C.4-6)}$$

The downrange equation of motion is

$$\frac{w}{g}\frac{d^2 x}{dt^2} = -\frac{\rho A V^2}{2} \left\{ C_{D_\infty} + \frac{K_D}{V^2} + (C_{N_\alpha} + K_{C_x}) \left[K_1^2 \, e^{\frac{2\lambda_1 x}{V_o}} + K_2^2 \, e^{\frac{2\lambda_2 x}{V_o}} + 2K_1 K_2 \, e^{\frac{2\lambda_o x}{V_o}} \cos\left(\frac{2\omega_o x}{V_o} + \nu_1 - \nu_2\right)\right]\right\} \quad \text{(C.4-7)}$$

which cannot be solved in closed form. As a result, perturbations are calculated for use as corrections. The perturbations are based on mean values of the drag coefficient with and without oscillations, where the mean drag coefficient is defined as

$$\overline{C_D} = \frac{\int_0^x C_D \, dx}{\int_0^x dx} \tag{C.4-8}$$

For the case without oscillations (zero angle of attack)

$$\overline{C_D}_{\alpha=0} = \frac{\int_0^x \left(C_{D_\infty} + \frac{K_D}{V^2}\right) dx}{\int_0^x dx} \tag{C.4-9}$$

The equation of motion becomes

$$\frac{w}{g}\frac{dV}{dt} = \frac{w}{g} V \frac{dV}{dx} = -\frac{\rho A}{2}\left(C_{D_\infty} + \frac{K_D}{V^2}\right) V^2 \tag{C.4-10}$$

or equivalently

$$\left(C_{D_\infty} + \frac{K_D}{V^2}\right) dx = -\frac{2w}{\rho A g}\frac{dV}{V} \tag{C.4-11}$$

Substituting Eq. (C.4-11) into the Eq. (C.4-9) and integrating yields

$$\overline{C_D}_{\alpha=0} = \frac{1}{x}\frac{2w}{\rho A g} \ln \frac{V_o}{V} \tag{C.4-12}$$

For the case with oscillatory motion, the mean drag coefficient is

$$\overline{C_D}_{\alpha} = \overline{C_D}_{\alpha=0} + \frac{C_{N_\alpha} + K_{C_x}}{x} \int_0^x \bar{\alpha}^2 \, dx \tag{C.4-13}$$

Upon substituting the expression for $\bar{\alpha}^2$, Eq. (C.4-7), and integrating

$$\overline{C_{D_\alpha}} = \overline{C_{D_{\alpha=0}}} + \frac{C_{N_\alpha} + K C_x}{x} \left\{ \frac{K_1^2 V_o}{2\lambda_1} \left[e^{\frac{2\lambda_1 x}{V_o}} - 1 \right] + \frac{K_2^2 V_o}{2\lambda_2} \left[e^{\frac{2\lambda_2 x}{V_o}} - 1 \right] \right.$$

$$\left. + \frac{V_o K_1 K_2}{\sqrt{\lambda_o^2 + \omega_o^2}} \left[e^{\frac{2\lambda_o x}{V_o}} \cos\left(\frac{2\omega_o x}{V_o} + \nu_1 - \nu_2 + \phi_o \right) - \cos(\nu_1 - \nu_2 + \phi_o) \right] \right\} \tag{C.4-14}$$

where

$$\phi_o = \sin^{-1} \frac{-\omega_o}{\sqrt{\lambda_o^2 + \omega_o^2}} = \cos^{-1} \frac{\lambda_o}{\sqrt{\lambda_o^2 + \omega_o^2}} \tag{C.4-15}$$

again taking note of the quadrant for ϕ_o .

The mean drag coefficients defined by Eqs. (C.4-12) and (C.4-14) are then used in constant drag coefficient equations for the motion:

$$V = V_o e^{-\frac{\rho A g}{2w} C_D x} \tag{C.4-16}$$

$$t = \frac{2w}{\rho A g} \frac{1}{V_o} \frac{1}{C_D} \left(e^{\frac{\rho A g}{2w} C_D x} - 1 \right) \tag{C.4-17}$$

The two results are subtracted to compute the perturbation quantities ΔV and Δt:

$$\Delta V = V_\alpha - V_{\alpha=0} = V_o \left[e^{-\frac{\rho A g}{2w} \overline{C_{D_\alpha}} x} - e^{-\frac{\rho A g}{2w} \overline{C_{D_{\alpha=0}}} x} \right] \tag{C.4-18}$$

$$\Delta t = t_\alpha - t_{\alpha=0} = \frac{2w}{\rho A g} \frac{1}{V_o} \left[\frac{e^{-\frac{\rho A g}{2w} \overline{C_{D_\alpha}} x} - 1}{\overline{C_{D_\alpha}}} - \frac{e^{-\frac{\rho A g}{2w} \overline{C_{D_{\alpha=0}}} x} - 1}{\overline{C_{D_{\alpha=0}}}} \right] \tag{C.4-19}$$

For values of range, X, prior to angle of attack convergence, these perturbations are applied directly to the velocity and flight time computed from the particle trajectory solution given in Section C.2, Eqs. (C.2-2) and (C.2-3). At values of range beyond the point of convergence, the particle trajectory solution from Eqs. (C.2-2) and (C.2-3) is restarted at the range of convergence, X_C. The initial conditions corresponding to the velocity and time at the convergence range, X_C, are the values for the particle trajectory equations (without angle of attack oscillations) perturbed by ΔV and Δt.

The range at convergence, X_C, is found by determining the point at which the angle of attack oscillations decay to a point where the drag is no longer appreciably affected by angle of attack. From Eq. (C.4-5), the upper envelope of the oscillations is given approximately by

$$\bar{\alpha}_{UPPER} \cong e^{\frac{\lambda_o x}{V_o}} \left[K_1 e^{\frac{\Delta\lambda}{V_o} x} + K_2 e^{-\frac{\Delta\lambda}{V_o} x} \right] \qquad \text{(C.4-20)}$$

The damping is assumed to be small compared to the natural frequency, so that many oscillations are required to reach half amplitude. For the purposes of numerical calculations, experience has shown pitch oscillations beyond the point where $\bar{\alpha}_{UPPER} = 0.5^o$ have little effect on the trajectory. Thus, the range at convergence is defined as the point at which this occurs. (This is a computer code input parameter and can be easily changed to any other value.)

The downrange error, ΔX, at nominal time, t_n, is computed by iterating the solution in range until the computed flight time matches t_n. If desired, the iteration can be avoided by an additional approximation. The downrange error is closely approximated by

$$\Delta x \cong (t_n - t) V \qquad \text{(C.4-21)}$$

C.4.2 Verification

Verification of the downrange perturbation equations was established by comparison to four numerical solutions of the full equations of motion given in Table C-3. The downrange position agrees to within 0.05%. The downrange perturbation, ΔX (the difference between the downrange position with oscillatory motion and the downrange position without oscillatory motion at the same flight time) agrees to approximately 10%.

APPENDIX D

HITS PROGRAM LISTING

This appendix presents the HITS[1] program listing. The code was designed to facilitate conversion to real-time operation from remote key board terminals. The program is written in FORTRAN IV and requires less than 175 K bytes of computer memory on an IBM 360.

The code generates seven (7) warning level (i.e., Level 4) diagnostics when compiled on an IBM-360/75: four (4) in Subroutine DOABC and three (3) in Subroutine DO234. These are to be expected and ignored.

---

[1] HITS is maintained in the Avco engineering computer code library as Production Code 5127.

D.1 MAIN Program

```
C                                                                       HITS0001
C                                                                       HITS0002
C ********************************************************************* HITS0003
C *                                                                   * HITS0004
C *                                                                   * HITS0005
C *   H      H  IIIIIII  TTTTTTT  SSSSSSS                             * HITS0006
C *   H      H     I        T     S                                   * HITS0007
C *   H      H     I        T     S                                   * HITS0008
C *   HHHHHHHH     I        T     SSSSSSS                             * HITS0009
C *   H      H     I        T           S                             * HITS0010
C *   H      H     I        T           S                             * HITS0011
C *   H      H  IIIIIII     T     SSSSSSS                             * HITS0012
C *                                                                   * HITS0013
C *                                                                   * HITS0014
C *                               CCCCCCC  OOOOOOO  DDDDDD   EEEEEEE  * HITS0015
C *                               C        O     O  D     D  E        * HITS0016
C *                               C        O     O  D     D  E        * HITS0017
C *                               C        O     O  D     D  EEEE     * HITS0018
C *                               C        O     O  D     D  E        * HITS0019
C *                               C        O     O  D     D  E        * HITS0020
C *                               CCCCCCC  OOOOOOO  DDDDDD   EEEEEEE  * HITS0021
C *                                                                   * HITS0022
C *                                                                   * HITS0023
C ********************************************************************* HITS0024
C *                                                                   * HITS0025
C *                                                                   * HITS0026
C *          HYPERVELOCITY INFLIGHT TRAJECTORY SCATTER CODE           * HITS0027
C *                                                                   * HITS0028
C *                                                                   * HITS0029
C *                     PREPARED BY                                   * HITS0030
C *                                                                   * HITS0031
C *                          AVCO SYSTEMS DIVISION                    * HITS0032
C *                          201 LOWELL STREET                        * HITS0033
C *                          WILMINGTON, MASSACHUSETTS 01887          * HITS0034
C *                                                                   * HITS0035
C *                                                                   * HITS0036
C *                     PREPARED FOR                                  * HITS0037
C *                                                                   * HITS0038
C *                          RESEARCH DIRECTORATE                     * HITS0039
C *                          GENERAL THOMAS J. RODMAN LABORATORY      * HITS0040
C *                          ROCK ISLAND ARSENAL                      * HITS0041
C *                          ROCK ISLAND, ILLINOIS 61201              * HITS0042
C *                                                                   * HITS0043
C *                                                                   * HITS0044
C *                     EFFECTIVE DATE - JANUARY 1976                 * HITS0045
C *                                                                   * HITS0046
C *                                                                   * HITS0047
C ********************************************************************* HITS0048
C                                                                       HITS0049
C                                                                       HITS0050
```

D.1 MAIN Program (continued)

```
      IMPLICIT REAL*8 (A-H,O-Y)
      LOGICAL*1   CARD(80)
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)
      COMMON /CINOM/ INOM
      COMMON /CISPRN/ ISPRNT
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2
      COMMON/HISPLT/ IOPLOT,ISPLOT,ZTITLE(20,30),ZDATAD(21,4,30)
      IOUT = 6
      INC = 5
C     INC = 10
C     REWIND INC
C     WRITE (IOUT,9001)
C9001 FORMAT('1 THE INPUT CARDS ...' // )
C   1 READ (5,9002,END=2)  CARD
C9002 FORMAT(80A1)
C     WRITE (INC,9002)  CARD
C     WRITE (IOUT,9003)  CARD
C9003 FORMAT(1X, 80A1)
C     GO TO 1
C   2 REWIND INC
    8 CONTINUE
      WRITE(IOUT,9005)
 9005 FORMAT(1H1,///,
     A 3(T9, 4('H'),T30, 4('H'),T40,25('I'),T71,25('T'),T102,25('S'),/),
     B10(T9, 4('H'),T30, 4('H'),T50, 4('I'),T81, 4('T'),T102, 4('S'),/),
     C 3(T9,25('H'),               T50, 4('I'),T81, 4('T'),T102,25('S'),/),
     D11(T9, 4('H'),T30, 4('H'),T50, 4('I'),T81, 4('T'),T123, 4('S'),/),
     E 3(T9, 4('H'),T30, 4('H'),T40,25('I'),T81, 4('T'),T102,25('S'),/),
     F1H0,T15,'HYPERVELOCITY',T48,'INFLIGHT',T78,'TRAJECTORY',T110,'SCAT
     GTER',/////,
     H1H0,   T55,'CCCCC  OOOOO  DDDD   EEEEE',/,
     I1H ,   T55,'C      O   O  D   D  E    ',/,
     J1H ,   T55,'C      O   O  D   D  EEE  ',/,
     K1H ,   T55,'C      O   O  D   D  E    ',/,
     L1H ,   T55,'CCCCC  OOOOO  DDDD   EEEEE',/,
     M///,T15,'PREPARED BY',T85,'PREPARED FOR',//,
     NT21,'AVCO SYSTEMS DIVISION',T91,'RESEARCH DIRECTORATE',/,
     OT21,'201 LOWELL STREET',T91,'GENERAL THOMAS J. RODMAN LABORATORY',
     P/,T21,'WILMINGTON, MASSACHUSETTS 01887',T91,'ROCK ISLAND ARSENAL',
     Q/,T91,'ROCK ISLAND, ILLINOIS 61201',////,
     RT53,'EFFECTIVE DATE - JANUARY 1976')
       DO 6 I=1,10
    6 ID(I,2) = 0
C INOM USED IN S2987.
      INOM = 1
C READ THE FIRST CARD AND PLACE THE PROPER DE SUBSCRIPTS INTO THE IA
C    ARRAY.
      CALL INITIL
```

D.1 MAIN Program (continued)

```
      IF ( MCOPT .EQ. 0 )   GO TO 9                                         HITS0101
      IF(IOPRNT.LT.3) IOPRNT=3                                              HITS0102
      IF ( MCALC .EQ. 3 )   IOPRNT = 4                                      HITS0103
    9 CONTINUE                                                              HITS0104
C READ THE OE NAME, TYPE, NOMINAL VALUE, TOLERANCE, VARIANCE AND TABLE      HITS0105
C    LENGTH AND PLACE INTO PROPER ARRAYS.                                   HITS0106
      CALL INCARD                                                           HITS0107
      KCSAVE = KC + 1                                                       HITS0108
C MOVE EITHER THE TYPE 1 OR TYPES 2, 3, AND 4 UP TO THE START OF THE        HITS0109
C    IA ARRAY.                                                              HITS0110
      CALL MOVEUP                                                           HITS0111
C                                                                           HITS0112
C  PRINT NAMES OF VARIABLES                                                 HITS0113
C                                                                           HITS0114
  520 FORMAT(1X)                                                            HITS0115
      IF(K234.EQ.0) GO TO 505                                               HITS0116
      WRITE(IOUT,501)                                                       HITS0117
  501 FORMAT(1H1,39X,'*** STOCHASTIC INDEPENDENT VARIABLES ***',//)         HITS0118
      DO 500 IK = 1,K234                                                    HITS0119
      IOES = IA(IK,1)                                                       HITS0120
      WRITE(IOUT,520)                                                       HITS0121
  500 CALL FILLIN(IOES,IOUT)                                                HITS0122
  505 CONTINUE                                                              HITS0123
      IF(KT1.EQ.0) GO TO 506                                                HITS0124
      WRITE(IOUT,507)                                                       HITS0125
  507 FORMAT(1H1,39X,'*** RANGE-CHECK INDEPENDENT VARIABLES ***',//)        HITS0126
      DO 508 IK = 1,KT1                                                     HITS0127
      IOES = IA(IK,1)                                                       HITS0128
      WRITE(IOUT,520)                                                       HITS0129
  508 CALL FILLIN(IOES,IOUT)                                                HITS0130
  506 CONTINUE                                                              HITS0131
      WRITE(IOUT,502)                                                       HITS0132
  502 FORMAT(1H1,47X,'*** DEPENDENT VARIABLES ***',//)                      HITS0133
      DO 503 IK = 1,KD                                                      HITS0134
      IOES = ID(IK,1)                                                       HITS0135
      WRITE(IOUT,520)                                                       HITS0136
  503 CALL FILLIN(IOES,IOUT)                                                HITS0137
      WRITE(IOUT,509)                                                       HITS0138
  509 FORMAT(1H1,39X,'*** VARIABLES RESET BY INPUT SEQUENCE ***',//)        HITS0139
      DO 510 IK = 1,200                                                     HITS0140
      IF(IA(IK,2).NE.-1) GO TO 510                                          HITS0141
      IOES = IA(IK,1)                                                       HITS0142
      WRITE(IOUT,520)                                                       HITS0143
      CALL FILLIN(IOES,IOUT)                                                HITS0144
  510 CONTINUE                                                              HITS0145
C CHECK COLUMN 2 OF IA FOR MISSING DATA.                                    HITS0146
      CALL CHCKIA                                                           HITS0147
C ARE WE RUNNING THE 3-DIM. MATRIX OR NON-TOL-VAR OR JUST NOMINAL VALUESHITS0148
      IF((KT1.NE.0).OR.(K234.NE.0)) CALL EXTRA                              HITS0149
      IF ( KT1 .EQ. 0 )   GO TO 10                                          HITS0150
```

D.1 MAIN Program (concluded)

```
      INOM = 0                                                          HITS0151
      CALL DOABC                                                        HITS0152
      GO TO 30                                                          HITS0153
   10 IF ( K234 .EQ. 0 )  GO TO 20                                      HITS0154
      ISAVIS = ISPRNT                                                   HITS0155
      ISPRNT = 0                                                        HITS0156
      CALL DO234                                                        HITS0157
      ISPRNT = ISAVIS                                                   HITS0158
      GO TO 30                                                          HITS0159
   20 CALL PICK1                                                        HITS0160
   30 CONTINUE                                                          HITS0161
      IF ( MCOPT .EQ. 0 )  GO TO 8                                      HITS0162
      KC = KCSAVE                                                       HITS0163
      CALL MCRL                                                         HITS0164
      IF(IOPLOT.EQ.0) GO TO 8                                           HITS0165
      CALL HYSPLT                                                       HITS0166
      GO TO 8                                                           HITS0167
      END                                                               HITS0168
```

D.2 Subroutine ARTLU

```
      SUBROUTINE ARTLU(/J/,/X/,/XT/,/YA/,/YAT/,/YB/,/YBT/,/YC/,/YCT/,  HITS0169
     1/YD/,/YDT/,/YE/,/YET/,/YF/,/YFT/,/YG/,/YGT/,/YH/,/YHT/,/YI/,/YIT/)HITS0170
        IMPLICIT REAL*8 (A-H,O-Z)                                       HITS0171
      DIMENSION XT(2),YAT(2),YBT(2),YCT(2),YDT(2),YET(2),YFT(2),YGT(2), HITS0172
     1YHT(2),YIT(2)                                                     HITS0173
      N=J                                                               HITS0174
      DO 100 I=1,4000                                                   HITS0175
      IF (X - XT(I+1)) 200,200,100                                      HITS0176
  200 P = (X-XT(I)) / (XT(I+1) - XT(I) )                                HITS0177
      GO TO (1,2,3,4,5,6,7,8,9,10), N                                   HITS0178
   10  CONTINUE                                                         HITS0179
    9 YI=YIT(I)+P*(YIT(I+1)-YIT(I))                                     HITS0180
    8 YH=YHT(I)+P*(YHT(I+1)-YHT(I))                                     HITS0181
    7 YG=YGT(I)+P*(YGT(I+1)-YGT(I))                                     HITS0182
    6 YF=YFT(I)+P*(YFT(I+1)-YFT(I))                                     HITS0183
    5 YE=YET(I)+P*(YET(I+1)-YET(I))                                     HITS0184
    4 YD=YDT(I)+P*(YDT(I+1)-YDT(I))                                     HITS0185
    3 YC=YCT(I)+P*(YCT(I+1)-YCT(I))                                     HITS0186
    2 YB=YBT(I)+P*(YBT(I+1)-YBT(I))                                     HITS0187
    1 YA=YAT(I)+P*(YAT(I+1)-YAT(I))                                     HITS0188
      GO TO 300                                                         HITS0189
  100 CONTINUE                                                          HITS0190
  300 RETURN                                                            HITS0191
      END                                                               HITS0192
```

D.3 Function A2

```
      FUNCTION A2   ( X )                                                HITS0193
           IMPLICIT REAL * 8 (A-H , O -Z )                               HITS0194
 COMMON/OEC/OE(600)                                                      HITS0195
 EQUIVALENCE              (V0      ,OE( 101)), (VWIND   ,OE( 102)),      HITS0196
A(PHIWIN ,OE( 103)), (RHO     ,OE( 104)), (CD1     ,OE( 105)),           HITS0197
B(V1     ,OE( 106)), (CD2     ,OE( 107)), (V2      ,OE( 108)),           HITS0198
C(XN     ,OE( 109)), (CNA     ,OE( 110)), (SM      ,OE( 111)),           HITS0199
D(CMQ    ,OE( 112)), (CMPA    ,OE( 113)), (TRIM    ,OE( 114)),           HITS0200
E(PNITRI ,OE( 115)), (ANGO    ,OE( 116)), (PHIANG  ,OE( 117)),           HITS0201
F(RATE   ,OE( 118)), (PHIRAT  ,OE( 119)), (GAMO    ,OE( 120)),           HITS0202
G(PHIGAM ,OE( 121)), (A       ,OE( 122)), (ELL     ,OE( 123)),           HITS0203
H(D      ,OE( 124)), (W       ,OE( 125)), (AIX     ,OE( 126)),           HITS0204
I(AIY    ,OE( 127)), (P       ,OE( 128)), (ALPCON  ,OE( 129)),           HITS0205
J(CAA    ,OE( 130))                                                      HITS0206
 EQUIVALENCE              (ALPHA   ,OE( 400)), (ATOT    ,OE( 401)),      HITS0207
A(ALPTRM ,OE( 402)), (ALPHDO  ,OE( 403)), (B       ,OE( 404)),           HITS0208
B(BETTRM ,OE( 405)), (BETA0   ,OE( 406)), (BETADO  ,OE( 407)),           HITS0209
C(BETA   ,OE( 408)), (ALPMAX  ,OE( 409)), (ALPMIN  ,OE( 410)),           HITS0210
D(CAYD   ,OE( 411)), (CD8     ,OE( 412)), (CMA     ,OE( 413)),           HITS0211
E(CMTHTD ,OE( 414)), (CMTHTA  ,OE( 415)), (CAY1    ,OE( 416)),           HITS0212
F(CAY2   ,OE( 417)), (CDAOB   ,OE( 418)), (CDAB    ,OE( 419)),           HITS0213
G(DELTT  ,OE( 420)), (DELV    ,OE( 421)), (DELT    ,OE( 422)),           HITS0214
H(DELX   ,OE( 423)), (DYDX0   ,OE( 424)), (DZDX0   ,OE( 425)),           HITS0215
I(DYDT0  ,OE( 426)), (DZDT0   ,OE( 427)), (DELW    ,OE( 428)),           HITS0216
J(DELLAM ,OE( 429)), (EYEP    ,OE( 430)), (EMP     ,OE( 431)),           HITS0217
K(EOLT   ,OE( 432)), (EDLT    ,OE( 433)), (F       ,OE( 434)),           HITS0218
L(H      ,OE( 435)), (PSID0   ,OE( 436)), (PHIO    ,OE( 437)),           HITS0219
M(PHI1   ,OE( 438)), (PHI2    ,OE( 439)), (PSIO    ,OE( 440)),           HITS0220
N(R1     ,OE( 441)), (R2      ,OE( 442)), (R3      ,OE( 443)),           HITS0221
O(R4     ,OE( 444)), (RTRIM   ,OE( 445)), (TC      ,OE( 446)),           HITS0222
P(TCO    ,OE( 447)), (TS      ,OE( 448)), (TGO     ,OE( 449)),           HITS0223
Q(TP     ,OE( 450)), (THETAO  ,OE( 451)), (THETDO  ,OE( 452)),           HITS0224
R(TOL    ,OE( 453)), (VC      ,OE( 454)), (VC0     ,OE( 455)),           HITS0225
S(VYA    ,OE( 456)), (VZA     ,OE( 457)), (WX      ,OE( 458))            HITS0226
 EQUIVALENCE              (WZ      ,OE( 459)), (WO      ,OE( 460)),      HITS0227
A(W1     ,OE( 461)), (W2      ,OE( 462)), (WL2     ,OE( 463)),           HITS0228
B(XLAM0  ,OE( 464)), (XLAM1   ,OE( 465)), (XLAM2   ,OE( 466)),           HITS0229
C(XNU1   ,OE( 467)), (XNU2    ,OE( 468)), (XG      ,OE( 469)),           HITS0230
D(XMU    ,OE( 470)), (XJAY    ,OE( 471)), (XJAZ    ,OE( 472)),           HITS0231
E(XJA    ,OE( 473)), (YA      ,OE( 474)), (ZA      ,OE( 475)),           HITS0232
F(ALPH   ,OE( 476))                                                      HITS0233
    XNNP  = XNU1 -XNU2 + PHIO                                            HITS0234
    A2    =  CAY1 * VO * (CAY1 /2.DO /XLAM1 * ( FEXP( 2.DO * XLAM1       HITS0235
1 * X / VO   )   -1.DO )  +  CAY2/ DSQRT( XLAM0 ** 2 + WO * WO ) *       HITS0236
2( DEXP(   2.DO * XLAMO/VO * X) * DCOS( 2.DO * WO * X /VO + XNNP )       HITS0237
3  - DCOS( XNNP)   )     ) + CAY2 * CAY2 * VO / XLAM2 *  ( FEXP( 2.DOHITS0238
4 * XLAM2 /VO   * X )    -1.DO  )                                        HITS0239
5  /2.DO                                                                 HITS0240
          RETURN                                                         HITS0241
      END                                                                HITS0242
```

D.4 Function A2F

```
      FUNCTION A2F ( X )                                                HITS0243
         IMPLICIT REAL * 8 (A-H , O -Z )                                HITS0244
       COMMON/OEC/OE(600)                                               HITS0245
       EQUIVALENCE            (VO      ,OE(    1)), (VWIND  ,OE(    2)), HITS0246
     A(PHIWIN ,OE(    3)), (RHO     ,OE(    4)), (CDI    ,OE(    5)),  HITS0247
     B(V1     ,OE(    6)), (CD2     ,OE(    7)), (V2     ,OE(    8)),  HITS0248
     C(XN     ,OE(    9)), (CNA     ,OE(   10)), (SM     ,OE(   11)),  HITS0249
     D(CMQ    ,OE(   12)), (CMPA    ,OE(   13)), (TRIM   ,OE(   14)),  HITS0250
     E(PNITRI ,OE(   15)), (ANGO    ,OE(   16)), (PHIANG ,OE(   17)),  HITS0251
     F(RATE   ,OE(   18)), (PHIRAT  ,OE(   19)), (GAMO   ,OE(   20)),  HITS0252
     G(PHIGAM ,OE(   21)), (A       ,OE(   22)), (ELL    ,OE(   23)),  HITS0253
     H(D      ,OE(   24)), (W       ,OE(   25)), (AIX    ,OE(   26)),  HITS0254
     I(AIY    ,OE(   27)), (P       ,OE(   28)), (ALPCON ,OE(   29)),  HITS0255
     J(CAA    ,OE(   30))                                               HITS0256
       EQUIVALENCE            (ALPHA   ,OE( 400)), (ATOT   ,OE( 401)),  HITS0257
     A(ALPTRM ,OE( 402)), (ALPHDO  ,OE( 403)), (B      ,OE( 404)),     HITS0258
     B(BETTRM ,OE( 405)), (BETAO   ,OE( 406)), (BETADO ,OE( 407)),     HITS0259
     C(BETA   ,OE( 408)), (ALPMAX  ,OE( 409)), (ALPMIN ,OE( 410)),     HITS0260
     D(CAYD   ,OE( 411)), (CD8     ,OE( 412)), (CMA    ,OE( 413)),     HITS0261
     E(CMTHTD ,OE( 414)), (CMTHTA  ,OE( 415)), (CAY1   ,OE( 416)),     HITS0262
     F(CAY2   ,OE( 417)), (CDAOB   ,OE( 418)), (CDAB   ,OE( 419)),     HITS0263
     G(DELTT  ,OE( 420)   (DELV    ,OE( 421)), (DELT   ,OE( 422)),     HITS0264
     H(DELX   ,OE( 423)   (YDXO    ,OE( 424)), (DZDXO  ,OE( 425)),     HITS0265
     I(DYDT   ,OE( 426)   (DZDTO   ,OE( 427)), (DELW   ,OE( 428)),     HITS0266
     J(DELLAM ,OE( 429) , (EYEP    ,OE( 430)), (EMP    ,OE( 431)),     HITS0267
     K(EOLT   ,OE( 432)), (EDLT    ,OE( 433)), (F      ,OE( 434)),     HITS0268
     L(H      ,OE( 435)), (PSIDO   ,OE( 436)), (PHIO   ,OE( 437)),     HITS0269
     M(PHI1   ,OE( 438)), (PHI2    ,OE( 439)), (PSIO   ,OE( 440)),     HITS0270
     N(R1     ,OE( 441)), (R2      ,OE( 442)), (R3     ,OE( 443)),     HITS0271
     O(R4     ,OE( 444)), (RTRIM   ,OE( 445)), (TC     ,OE( 446)),     HITS0272
     P(TCO    ,OE( 447)), (TS      ,OE( 448)), (TGO    ,OE( 449)),     HITS0273
     Q(TP     ,OE( 450)), (THETAO  ,OE( 451)), (THETDO ,OE( 452)),     HITS0274
     R(TOL    ,OE( 453)), (VC      ,OE( 454)), (VCO    ,OE( 455)),     HITS0275
     S(VYA    ,OE( 456)), (VZA     ,OE( 457)), (WX     ,OE( 458))      HITS0276
       EQUIVALENCE            (WZ      ,OE( 459)), (WO     ,OE( 460)),  HITS0277
     A(W1     ,OE( 461)), (W2      ,OE( 462)), (WL2    ,OE( 463)),     HITS0278
     B(XLAMO  ,OE( 464)), (XLAM1   ,OE( 465)), (XLAM2  ,OE( 466)),     HITS0279
     C(XNU1   ,OE( 467)), (XNU2    ,OE( 468)), (XG     ,OE( 469)),     HITS0280
     D(XMU    ,OE( 470)), (XJAY    ,OE( 471)), (XJAZ   ,OE( 472)),     HITS0281
     E(XJA    ,OE( 473)), (YA      ,OE( 474)), (ZA     ,OE( 475)),     HITS0282
     F(ALPH   ,OE( 476))                                               HITS0283
        XNNP = XNU1 -XNU2 + PHIO                                       HITS0284
       A2F  =  CAY1 * VO * (CAY1 /2.DO /XLAM1 * ( FEXP( 2.DO * XLAM1   HITS0285
     1 * X / VO  )  -1.DO )  +  CAY2/ DSQRT( XLAMO ** 2 + WO * WO ) *  HITS0286
     2( DEXP(  2.DO * XLAMO/VO * X) * DCOS( 2.DO * WO * X /VO + XNNP ) HITS0287
     3  - DCOS( XNNP)  )    ) + CAY2 * CAY2 * VO / XLAM2 *  ( FEXP( 2.DOHITS0288
     4 * XLAM2 /VO   * X )   -1.DO   )                                 HITS0289
     5  /2.DC                                                          HITS0290
            RETURN                                                     HITS0291
          END                                                          HITS0292
```

D.5 Block Data

```
      BLOCK DATA                                                        HITS0293
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0294
      COMMON/NAMES/HEAD(180)                                            HITS0295
      DIMENSION HEAD1(132),HEAD2(48)                                    HITS0296
      EQUIVALENCE (HEAD1(1), HEAD(1)), (HEAD2(1), HEAD(133))            HITS0297
      DATA HEAD1      /'    V0','    WX','    WZ','   RHO','   CX1',    HITS0298
     A'    V1','   CX2','    V2','    XN','   CNA','    SM','   CM0',    HITS0299
     B'  CMPA',' ATRMS',' BTRMS',' THETO','  PSIO',' TDOTO',' PDOTO',   HITS0300
     C'  GAMO','   AZO','     A','   ELL','     D','     W','   AIX',    HITS0301
     D'   AIY','     P','ALPCON','   CAA',                              HITS0302
     E' XBART',' YBART',' ZBART',' YBARX',' ZBARX',                     HITS0303
     F' IFCRW',' IFCRC','   IFC',' IDUMP',' ICONV',' ICNCL',            HITS0304
     G'XBARTI','YBARTI','ZBARTI','YBARXI','ZBARXI','   DXT','   DYT',   HITS0305
     H'   DZT','   DYX','   DZX','  DVXT','  DVYT','  DVZT','  DVXX',   HITS0306
     I'  DVYX','  DVZX','    DT','  RADT','  RADX','  RHOT','  RHOX',   HITS0307
     J'  DXTI','  DYTI','  DZTI','  DXTE','  DYTE','  DZTE','  DYXI',   HITS0308
     K'  DZXI','  DYXE','  DZXE',' DXDYT',' DXDZT',' DYDZT',' DYDZX',   HITS0309
     L'DXDYTI','DXDZTI','DYDZTI','DYDZXI','DXDYTE','DXDZTE','DYDZTE',   HITS0310
     M'DYDZXE',' ALPHA','  ATOT','ALPTRM','ALPHDO','     B','BETTRM',   HITS0311
     N' BETAO','BETADO','  BETA','ALPMAX','ALPMIN','  CAYD','   CDB',   HITS0312
     O'   CMA','CMTHTD','CMTHTA','  CAY1','  CAY2',' CDAOB','  CDAB',   HITS0313
     P' DELTT','  DELV','  DELT','  DELX',' DYDXO',' DZDXO',' DYDTO',   HITS0314
     Q' DZDTO','  DELW','DELLAM','  EYEP','   EMP','  EOLT','  EDLT',   HITS0315
     R'     F','     H',' PSIDO','  PHIO','  PHI1','  PHI2','  PSIO',   HITS0316
     S'    R1','    R2','    R3','    R4',' RTRIM','    TC','   TCO'/   HITS0317
      DATA HEAD2      /'    TS','   TGO','    TP','THETAO','THETDO',    HITS0318
     A'   TOL','    VC','   VCO','   VYA','   VZA','    WX','    WZ',   HITS0319
     B'    W0','    W1','    W2','   WL2',' XLAMO',' XLAM1',' XLAM2',   HITS0320
     C'  XNU1','  XNU2','    XG','   XMU','  XJAY','  XJAZ','   XJA',   HITS0321
     D'    YA','    ZA',' ALPHO',                                       HITS0322
     E'    TN','  XTFC','  YTFC','  ZTFC',' VXTFC',' VYTFC',' VZTFC',   HITS0323
     F'    XT','    YT','    ZT','   VXT','   VYT','   VZT','    YR',   HITS0324
     G'    ZR','   VXR','   VYR','   VZR','    TX'/                     HITS0325
      END                                                               HITS0326
```

D.6 Subroutine CHCKIA

```
      SUBROUTINE CHCKIA                                                 HITS0327
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0328
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,       HITS0329
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS0330
      WRITE(IOUT,8000)                                                  HITS0331
 8000 FORMAT(1H1,40X,40H*** VARIABLES ASSIGNED PRESET VALUES ***,//)    HITS0332
      JKOUT = IOUT                                                      HITS0333
C IF COLUMN 2 OF IA HAS A ZERO, WE MUST FILL IN SOME DATA.              HITS0334
      DO 10 I=1,KA                                                      HITS0335
      IAT = IA(I,1)                                                     HITS0336
      IF ( IA(I,2) .EQ. 0 )  CALL FILLIN(IAT,JKOUT)                     HITS0337
   10 CONTINUE                                                          HITS0338
      WRITE(IOUT,8001)                                                  HITS0339
 8001 FORMAT(1H1)                                                       HITS0340
      RETURN                                                            HITS0341
      END                                                               HITS0342
```

D.7 Subroutine CHCKIN

```
      SUBROUTINE CHCKIN                                                 HITS0343
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0344
      COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS0345
     1IY,KARS1(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS0346
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS0347
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS0348
      DIMENSION IB(2,200)                                               HITS0349
      EQUIVALENCE (IB(1,1),B(1))                                        HITS0350
C      THIS SUBROUTINE CHECKS THE INPUTS AND                            HITS0351
C      ...TYPE 8 COUNTS THE VARIABLES                                   HITS0352
C      ...TYPE 2 FORCES THE TOLERANCE TO BE A 3 SIGMA MINIMUM           HITS0353
C      ...TYPE 4   DEFINES TOLERANCE=.5(XU-XL), MEAN=.5(XU+XL) AND PUTS HITS0354
C                  THE MAXIMUM VALUE OF THE TAB. DIST. FUNCT. IN THE VARIHITS0355
      IN8=0                                                             HITS0356
      DO 2 I=1,KD                                                       HITS0357
      IF(ID(I,2))2,2,1                                                  HITS0358
    1  IN8=IN8+1                                                        HITS0359
    2  CONTINUE                                                         HITS0360
      DO 9 I=1,K234                                                     HITS0361
      IADD=IA(I,2)                                                      HITS0362
      ITYP=IB(1,IADD)                                                   HITS0363
      IF(ITYP-3)3,9,5                                                   HITS0364
C      FOR ITYP=2 IT IS GAUSSIAN IT,IV=B ADDRESSES OF TOL AND VAR       HITS0365
    3 IT=IADD+2                                                         HITS0366
      IV=IT+1                                                           HITS0367
      TOL=3.*DSQRT(B(IV))                                               HITS0368
      IF(TOL-B(IT))9,9,4                                                HITS0369
    4  B(IT)=TOL                                                        HITS0370
      GO TO 9                                                           HITS0371
C      TYPE 4 DISTRIBUTED VARIABLE ENTRY                                HITS0372
    5  VAR=0.                                                           HITS0373
      IT=IADD+4                                                         HITS0374
      ILT=IB(1,IT)                                                      HITS0375
      IT=IT+1                                                           HITS0376
      IS=IB(1,IT)                                                       HITS0377
      IV=ILT+IS-1                                                       HITS0378
C      IT IS THE ADDRESS IN B OF THE 1ST C ADDRESS FOR THE TAB. DIST. FCHITS0379
C      ILT IS THE NUMBER OF ENTRIES IN C                                HITS0380
C      IV IS THE LAST ADDRESS IN C OF THE DIST. FUNCT.                  HITS0381
      DO 6 J=IS,IV                                                      HITS0382
      IF(VAR-C(J))7,6,6                                                 HITS0383
    7  VAR=C(J)                                                         HITS0384
    6  CONTINUE                                                         HITS0385
C      VAR IS THE MAX. VALUE STORE IN VARIANCE POSITION OF B ENTRYS     HITS0386
C      CALCULATE 1ST AND LAST ADDRESSES OF VARIABLE TABLE IN C          HITS0387
      IT=IV+1                                                           HITS0388
      IV=IT+ILT-1                                                       HITS0389
      IADD=IADD+1                                                       HITS0390
      B(IADD)=.5*(C(IT)+C(IV))                                          HITS0391
      IADD=IADD+1                                                       HITS0392
```

D.7 Subroutine CHCKIN (concluded)

```
      B(IADD)=.5*(C(IV)-C(IT))                                          HITS0393
      IADD=IADD+1                                                       HITS0394
      B(IADD)=VAR                                                       HITS0395
    9  CONTINUE                                                         HITS0396
      RETURN                                                            HITS0397
      END                                                               HITS0398
```

D.8 Subroutine CNVRT

```
      SUBROUTINE CNVRT(RN)                                              HITS0399
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0400
      DIMENSION R(11)                                                   HITS0401
      DATA R/0.[illegible],0.28,0.36,0.41,0.46,0.5,0.54,0.59,0.64,0.72,1.0/   HITS0402
C      THIS PROGRAM CONVERTS RECT. DIST. NOS. TO GAUSSIAN DIST. NOS.    HITS0403
      AJ=10.*RN                                                         HITS0404
      J=AJ                                                              HITS0405
      BJ=J                                                              HITS0406
      RN=(AJ-BJ)*(R(J+2)-R(J+1))+R(J+1)                                 HITS0407
      RETURN                                                            HITS0408
      END                                                               HITS0409
```

D.9 Subroutine CROSS

```
      SUBROUTINE  CROSS( X , V  )                                       HITS0410
        IMPLICIT REAL *8 (A-H, O -Z )                                   HITS0411
       COMMON/OEC/OE(600)                                               HITS0412
       EQUIVALENCE             (VO       ,OE( 101)), (VWIND   ,OE( 102)),HITS0413
     A(PHIMIN  ,OE( 103)), (RHO      ,OE( 104)), (CDI     ,OE( 105)),   HITS0414
     B(VI      ,OE( 106)), (CD2      ,OE( 107)), (V2      ,OE( 108)),   HITS0415
     C(XN      ,OE( 109)), (CNA      ,OE( 110)), (SM      ,OE( 111)),   HITS0416
     D(CMQ     ,OE( 112)), (CMPA     ,OE( 113)), (TRIM    ,OE( 114)),   HITS0417
     E(PNITPT  ,OE( 115)), (ANGO     ,OE( 116)), (PHIANG  ,OE( 117)),   HITS0418
     F(RATE    ,OE( 118)), (PHIRAT   ,OE( 119)), (GAMO    ,OE( 120)),   HITS0419
     G(PHIGAM  ,OE( 121)), (A        ,OE( 122)), (FLL     ,OE( 123)),   HITS0420
     H(D       ,OE( 124)), (W        ,OE( 125)), (AIX     ,OE( 126)),   HITS0421
     I(AIY     ,OE( 127)), (P        ,OE( 128)), (ALPCON  ,OE( 129)),   HITS0422
     J(CAA     ,OE( 130))                                               HITS0423
       EQUIVALENCE             (IDUMP    ,OE( 204))                     HITS0424
       EQUIVALENCE             (ALPHA    ,OE( 400)), (ATOT    ,OE( 401)),HITS0425
     A(ALPTRM  ,OE( 402)), (ALPHOO   ,OE( 403)), (B       ,OE( 404)),   HITS0426
     B(BETTRM  ,OE( 405)), (BETAO    ,OE( 406)), (BETADO  ,OE( 407)),   HITS0427
     C(BETA    ,OE( 408)), (ALPMAX   ,OE( 409)), (ALPMIN  ,OE( 410)),   HITS0428
     D(CAYD    ,OE( 411)), (CD8      ,OE( 412)), (CMA     ,OE( 413)),   HITS0429
     E(CMTHTD  ,OE( 414)), (CMTHTA   ,OE( 415)), (CAY1    ,OE( 416)),   HITS0430
     F(CAY2    ,OE( 417)), (CDAOB    ,OE( 418)), (CDAB    ,OE( 419)),   HITS0431
     G(DELTT   ,OE( 420)), (DELV     ,OE( 421)), (DELT    ,OE( 422)),   HITS0432
     H(DELX    ,OE( 423)), (DYDXO    ,OE( 424)), (DZDXO   ,OE( 425)),   HITS0433
     I(DYDT    ,OE( 426)), (DZDTO    ,OE( 427)), (DELW    ,OE( 428)),   HITS0434
     J(DELLAM  ,OE( 429)), (EYEP     ,OE( 430)), (EMP     ,OE( 431)),   HITS0435
     K(EOLT    ,OE( 432)), (EDLT     ,OE( 433)), (F       ,OE( 434)),   HITS0436
     L(H       ,OE( 435)), (PSIDO    ,OE( 436)), (PHIO    ,OE( 437)),   HITS0437
     M(PHI1    ,OE( 438)), (PHI2     ,OE( 439)), (PSIO    ,OE( 440)),   HITS0438
     N(R1      ,OE( 441)), (R2       ,OE( 442)), (R3      ,OE( 443)),   HITS0439
     O(R4      ,OE( 444)), (RTRIM    ,OE( 445)), (TC      ,OE( 446)),   HITS0440
     P(TCO     ,OE( 447)), (TS       ,OE( 448)), (TGO     ,OE( 449)),   HITS0441
     Q(TP      ,OE( 450)), (THETAO   ,OE( 451)), (THETDO  ,OE( 452)),   HITS0442
     R(TOL     ,OE( 453)), (VC       ,OE( 454)), (VCO     ,OE( 455)),   HITS0443
     S(VYA     ,OE( 456)), (VZA      ,OE( 457)), (WX      ,OE( 458))    HITS0444
       EQUIVALENCE             (WZ       ,OE( 459)), (WO      ,OE( 460)),HITS0445
     A(W1      ,OE( 461)), (W2       ,OE( 462)), (WL2     ,OE( 463)),   HITS0446
     B(XLAMO   ,OE( 464)), (XLAM1    ,OE( 465)), (XLAM2   ,OE( 466)),   HITS0447
     C(XNU1    ,OE( 467)), (XNU2     ,OE( 468)), (XG      ,OE( 469)),   HITS0448
     D(XMU     ,OE( 470)), (XJAY     ,OE( 471)), (XJAZ    ,OE( 472)),   HITS0449
     E(XJA     ,OE( 473)), (YA       ,OE( 474)), (ZA      ,OE( 475)),   HITS0450
     F(ALPHO   ,OE( 476))                                               HITS0451
         TP   = X/VO                                                    HITS0452
         TL1  = XLAM1  *  TP                                            HITS0453
         TL2  = XLAM2  *  TP                                            HITS0454
         ELT1    =           FEXP( TL1 )                                HITS0455
         ELT2    =           FEXP( TL2 )                                HITS0456
         W1T     =    W1  *  TP                                         HITS0457
         W1TV    =    W1T        + XNU1                                 HITS0458
         W2T     =    W2  *  TP                                         HITS0459
```

D.9 Subroutine CROSS (continued)

```
          W2TV  =  W2T       - XNU2                                                HITS0460
 1000  FORMAT(    '      TP          ALPHA          BETA         ATOT          VCOHITS0461
      1VY           VCONZ         YCONA         ZCONA          VYA          VZA'   HITS0462
      2/1X, 1P10E12.4    // )                                                      HITS0463
          ALPHA   =ELT1*CAY1*DSIN( W1TV )- ELT2*CAY2                               HITS0464
      1                                         * DSIN(W2TV)+ ALPTRM               HITS0465
          BETA    =ELT1*CAY1*DCOS( W1TV )+ ELT2*CAY2                               HITS0466
      1                                         * DCOS(W2TV)+ BETTRM               HITS0467
              TP2 = TP * TP                                                        HITS0468
        A2 =0.D0                                                                   HITS0469
         B2=0.D0                                                                   HITS0470
         IF( DABS( ALPHA) .GT.  1.D-38)                                            HITS0471
      1  A2=    ALPHA*ALPHA                                                        HITS0472
          IF( DABS(BETA) .GT. 1.D-38)  B2=BETA*BETA                                HITS0473
        ATOT= DSQRT( A2+B2)                                                        HITS0474
             P2 = P * P                                                            HITS0475
             PTP =P*TP                                                             HITS0476
          IF(  P .EQ. 0.)  GO  TO  1                                               HITS0477
           SINPT =  DSIN( PTP )                                                    HITS0478
           COSPT =  DCOS( PTP )                                                    HITS0479
          VCONY= (ALPTRM * SINPT -BETTRM *(COSPT -1.D0))/P                         HITS0480
          VCONZ= -( BETTRM * SINPT + ALFTRM *(COSPT -1.D0)) /P                     HITS0481
         YCONA  =  ( ALPTRM * (1.D0 - COSPT) +BETTRM*(PTP-SINPT))/ P2              HITS0482
         ZCONA  =  (-BETTRM * (1.D0 - COSPT) +ALPTRM*(PTP-SINPT))/ P2              HITS0483
          GO TO  2                                                                 HITS0484
    1    CONTINUE                                                                  HITS0485
            VCONY = TP * ALPTRM                                                    HITS0486
            VCONZ = -BETTRM * TP                                                   HITS0487
            YCONA =  TP2 * ALPTRM /2.D0                                            HITS0488
            ZCONA = -TP2 * BETTRM /2.D0                                            HITS0489
    2    CONTINUE                                                                  HITS0490
           HV   = H       /V0                                                      HITS0491
           HVV = HV * V                                                            HITS0492
         ELWP1  =  XLAM1 ** 2  +(W1 + P)** 2                                       HITS0493
         ELWP2  =  XLAM2 ** 2  +(W2 - P)** 2                                       HITS0494
         CAY1LW =  CAY1 / DSQRT( ELWP1)                                            HITS0495
         CAY2LW =  CAY2 / DSQRT( ELWP2)                                            HITS0496
          XNUPH1 =  XNU1 + PHI1                                                    HITS0497
          XNUPH2 =-XNU2 + PHI2                                                     HITS0498
          WPTVP1 =(W1 + P) * TP  + XNUPH1                                          HITS0499
          WPTVP2 =(W2 - P) * TP  + XNUPH2                                          HITS0500
         COSNP1 =  DCOS(XNUPH1)                                                    HITS0501
         COSNP2 =  DCOS(XNUPH2)                                                    HITS0502
          SINNP1 = DSIN( XNUPH1)                                                   HITS0503
          SINNP2 = DSIN( XNUPH2)                                                   HITS0504
         VYA = HVV *( CAY1LW *(ELT1 * DSIN( WPTVP1)- SINNP1        )               HITS0505
      1               -CAY2LW *(ELT2 * DSIN( WPTVP2)- SINNP2        )              HITS0506
      2 + VCONY  + DYDT0/ H )                                                      HITS0507
         VZA = HVV *(-CAY1LW *(ELT1 * DCOS( WPTVP1)- COSNP1        )               HITS0508
      1               -CAY2LW *(ELT2 * DCOS( WPTVP2)- COSNP2        )              HITS0509
```

D.9 Subroutine CROSS (concluded)

```
2 + VCONZ   + DZDTO/ H )                                                HITS0510
     WCON1  = WPTVP1 + PHI1                                             HITS0511
     WCON2  = WPTVP2 + PHI2                                             HITS0512
  IF( IDUMP     .EQ.  1 ) WRITE(6,1000) TP .ALPHA .BETA . ATOT          HITS0513
1.VCONY .VCONZ   .   YCONA . ZCONA   . VYA . VZA                        HITS0514
    YA   = H *(   CAY1/ ELWP1 * ELT1 * DSIN(WCON1 )-CAY2/ELWP2* ELT2 HITS0515
1 *DSIN(WCON2)- (CAY1LW * DSIN( XNUPH1) -CAY2LW * DSIN(XNUPH2)-DYDTHITS0515
20 /H ) * TP    -CAY1/ ELWP1 * DSIN(XNUPH1+PHI1) +CAY2/ELWP2 * DSIN HITS0517
3(XNUPH2 +PHI2)  + YCONA   )                                            HITS0518
    ZA =  H*(-CAY1/ELWP1*    ELT1   * DCOS(WCON1 )-CAY2/ELWP2* ELT2 HITS0519
1 *DCOS(WCON2)+ (CAY1LW * DCOS(XNUPH1  ) +CAY2LW * DCOS(XNUPH2)+DZDTHITS0520
20 /H ) * TP    + CAY1/ELWP1   * DCOS(XNUPH1+PHI1) +CAY2/ELWP2 * DCOS HITS0521
3(XNUPH2 +PHI2) + ZCONA   )                                             HITS0522
       EOLT =  EEXP( XLAMO * TP)                                        HITS0523
       EDLT =  DEXP( DELLAM* TP)                                        HITS0524
     ALPMAX =  EOLT *( CAY1 * EOLT   + CAY2 / EDLT )                    HITS0525
     ALPMIN =  EOLT *( CAY1 * EOLT   - CAY2 / EDLT )                    HITS0526
     XJAY = -HV * ( CAY1LW * SINNP1 -CAY2LW * SINNP2   )*1.D3           HITS0527
     XJAZ =  HV * ( CAY1LW * COSNP1 +CAY2LW * COSNP2   )*1.D3           HITS0528
      XJA = DSQRT( XJAY ** 2  + XJAZ ** 2 )                             HITS0529
      IF(IDUMP    ,EQ. 1 ) WRITE(6,1001) YA , ZA, EOLT , EDLT           HITS0530
1 , ALPMAX , ALPMIN , XJAY , XJAZ ,   XJA                               HITS0531
1001  FORMAT(  '        YA           ZA           EDLT         EDLT       ALPHITS0532
1MAX      ALPMIN        XJAY         XJAZ          XJA  ' /1X,1P11E12.HITS0533
24 // )                                                                 HITS0534
          RETURN                                                        HITS0535
           END                                                          HITS0536
```

D.10 Subroutine CROSSF

```
      SUBROUTINE  CROSSF( X , V  )                                     HITS0537
         IMPLICIT REAL *8 (A-H, O -Z )                                 HITS0538
     COMMON/OEC/OE(600)                                                HITS0539
     EQUIVALENCE         (VO     ,OE(   1)), (VWIND   ,OE(   2)),      HITS0540
    A(PHIWIN ,OE(   3)), (RHO    ,OE(   4)), (CD1     ,OE(   5)),      HITS0541
    B(V1     ,OE(   6)), (CD2    ,OE(   7)), (V2      ,OE(   8)),      HITS0542
    C(XN     ,OE(   9)), (CNA    ,OE(  10)), (SM      ,OE(  11)),      HITS0543
    D(CMQ    ,OE(  12)), (CMPA   ,OE(  13)), (TRIM    ,OE(  14)),      HITS0544
    E(PNITPI ,OE(  15)), (ANGO   ,OE(  16)), (PHIANG  ,OE(  17)),      HITS0545
    F(RATE   ,OE(  18)), (PHIRAT ,OE(  19)), (GAMO    ,OE(  20)),      HITS0546
    G(PHIGAM ,OE(  21)), (A      ,OE(  22)), (ELL     ,OE(  23)),      HITS0547
    H(D      ,OE(  24)), (W      ,OE(  25)), (AIX     ,OE(  26)),      HITS0548
    I(AIY    ,OE(  27)), (P      ,OE(  28)), (ALPCON  ,OE(  29)),      HITS0549
    J(CAA    ,OE(  30))                                                HITS0550
     EQUIVALENCE         (IDUMP  ,OE( 204))                            HITS0551
     EQUIVALENCE         (ALPHA  ,OE( 400)), (ATOT    ,OE( 401)),      HITS0552
    A(ALPTRM ,OE( 402)), (ALPHDO ,OE( 403)), (B       ,OE( 404)),      HITS0553
    B(BETTRM ,OE( 405)), (BETAO  ,OE( 406)), (BETADO  ,OE( 407)),      HITS0554
    C(BETA   ,OE( 408)), (ALPMAX ,OE( 409)), (ALPMIN  ,OE( 410)),      HITS0555
    D(CAYD   ,OE( 411)), (CD8    ,OE( 412)), (CMA     ,OE( 413)),      HITS0556
    E(CMTHTD ,OE( 414)), (CMTHTA ,OE( 415)), (CAY1    ,OE( 416)),      HITS0557
    F(CAY2   ,OE( 417)), (CDA08  ,OE( 418)), (CDAB    ,OE( 419)),      HITS0558
    G(DELTT  ,OE( 420)), (DELV   ,OE( 421)), (DELT    ,OE( 422)),      HITS0559
    H(DELX   ,OE( 423)), (DYDXO  ,OE( 424)), (DZDXO   ,OE( 425)),      HITS0560
    I(DYDT   ,OE( 426)), (DZDTO  ,OE( 427)), (DELW    ,OE( 428)),      HITS0561
    J(DELLAM ,OE( 429)), (EYEP   ,OE( 430)), (EMP     ,OE( 431)),      HITS0562
    K(ECLT   ,OE( 432)), (EDLT   ,OE( 433)), (F       ,OE( 434)),      HITS0563
    L(H      ,OE( 435)), (PSIDO  ,OE( 436)), (PHIO    ,OE( 437)),      HITS0564
    M(PHI1   ,OE( 438)), (PHI2   ,OE( 439)), (PSIO    ,OE( 440)),      HITS0565
    N(R1     ,OE( 441)), (R2     ,OE( 442)), (R3      ,OE( 443)),      HITS0566
    O(R4     ,OE( 444)), (RTRIM  ,OE( 445)), (TC      ,OE( 446)),      HITS0567
    P(TCO    ,OE( 447)), (TS     ,OE( 448)), (TGO     ,OE( 449)),      HITS0568
    Q(TP     ,OE( 450)), (THETAO ,OE( 451)), (THETDO  ,OE( 452)),      HITS0569
    R(TOL    ,OE( 453)), (VC     ,OE( 454)), (VCO     ,OE( 455)),      HITS0570
    S(VYA    ,OE( 456)), (VZA    ,OE( 457)), (WX      ,OE( 458))       HITS0571
     EQUIVALENCE         (WZ     ,OE( 459)), (WO      ,OE( 460)),      HITS0572
    A(W1     ,OE( 461)), (W2     ,OE( 462)), (WL2     ,OE( 463)),      HITS0573
    B(XLAM   ,OE( 464)), (XLAM1  ,OE( 465)), (XLAM2   ,OE( 466)),      HITS0574
    C(XNU1   ,OE( 467)), (XNU2   ,OE( 468)), (XG      ,OE( 469)),      HITS0575
    D(XMU    ,OE( 470)), (XJAY   ,OE( 471)), (XJAZ    ,OE( 472)),      HITS0576
    E(XJA    ,OE( 473)), (YA     ,OE( 474)), (ZA      ,OE( 475)),      HITS0577
    F(ALPH   ,OE( 476))                                                HITS0578
       TP  = X/VO                                                      HITS0579
       TL1 = XLAM1 * TP                                                HITS0580
       TL2 = XLAM2 * TP                                                HITS0581
       ELT1   =          FEXP( TL1 )                                   HITS0582
       ELT2   =          FEXP( TL2 )                                   HITS0583
       W1T    =   W1 * TP                                              HITS0584
       W1TV   =   W1T        + XNU1                                    HITS0585
       W2T    =   W2 * TP                                              HITS0586
```

D.10 Subroutine CROSSF (continued)

```
        W2TV  =  W2T       - XNU2                                         HITS0587
1000  FORMAT(   '      TP          ALPHA          BETA          ATOT     VCOHITS0588
     1NY         VCONZ         YCONA          ZCONA          VYA        VZA'  HITS0589
     2/1X, 1P10E12.4    // )                                              HITS0590
                                                                          HITS0591
      ALPHA   =ELT1*CAY1*DSIN( W1TV )- ELT2*CAY2                          HITS0592
     1                                    * DSIN(W2TV)+ ALPTRM            HITS0593
      BETA    =ELT1*CAY1*DCOS( W1TV )+ ELT2*CAY2                          HITS0594
     1                                    * DCOS(W2TV)+ BETTRM            HITS0595
          TP2 = TP * TP                                                   HITS0596
   A2 =0.D0                                                               HITS0597
     B2=0.D0                                                              HITS0598
     IF( DABS( ALPHA) .GT.  1.D-38)                                       HITS0599
     1  A2=     ALPHA*ALPHA                                               HITS0600
      IF( DABS(BETA) .GT. 1.D-38)  B2=BETA*BETA                           HITS0601
    ATOT= DSQRT( A2+B2)                                                   HITS0602
          P2 = P * P                                                      HITS0603
          PTP =P*TP                                                       HITS0604
       IF(  P .EQ. 0.)  GO  TO  1                                         HITS0605
        SINPT =  DSIN( PTP )                                              HITS0606
        COSPT =  DCOS( PTP )                                              HITS0607
       VCONY= (ALPTRM * SINPT -BETTRM *(COSPT -1.D0))/P                   HITS0608
       VCONZ= -( BETTRM * SINPT + ALPTRM *(COSPT -1.D0)) /P               HITS0609
      YCONA  =   ( ALPTRM * (1.D0 - COSPT) +BETTRM*(PTP-SINPT))/ P2       HITS0610
      ZCONA  =   (-BETTRM * (1.D0 - COSPT) +ALPTRM*(PTP-SINPT))/ P2       HITS0611
       GO TO  2                                                           HITS0612
1     CONTINUE                                                            HITS0613
         VCONY = TP * ALPTRM                                              HITS0614
         VCONZ = -BETTRM * TP                                             HITS0615
         YCONA =  TP2 * ALPTRM /2.D0                                      HITS0616
         ZCONA = -TP2 * BETTRM /2.D0                                      HITS0617
2     CONTINUE                                                            HITS0618
        HV   = H         /V0                                              HITS0619
        HVV  = HV * V                                                     HITS0620
     ELWP1   =   XLAM1 ** 2  +(W1 + P)** 2                                HITS0621
     ELWP2   =   XLAM2 ** 2  +(W2 - P)** 2                                HITS0622
     CAY1LW  =   CAY1 / DSQRT( ELWP1)                                     HITS0623
     CAY2LW  =   CAY2 / DSQRT( ELWP2)                                     HITS0624
       XNUPH1 = XNU1 + PHI1                                               HITS0625
       XNUPH2 =-XNU2 + PHI2                                               HITS0626
       WPTVP1 =(W1 + P) * TP  + XNUPH1                                    HITS0627
       WPTVP2 =(W2 - P) * TP  + XNUPH2                                    HITS0628
     COSNP1 =  DCOS(XNUPH1)                                               HITS0629
     COSNP2 =  DCOS(XNUPH2)                                               HITS0630
       SINNP1 = DSIN( XNUPH1)                                             HITS0631
       SINNP2 = DSIN( XNUPH2)                                             HITS0632
     VYA = HVV *( CAY1LW *(ELT1 * DSIN( WPTVP1)- SINNP1           )       HITS0633
     1            -CAY2LW *(ELT2 * DSIN( WPTVP2)- SINNP2           )       HITS0634
     2 + VCONY  + DYDTC/ H )                                              HITS0635
     VZA = HVV *(-CAY1LW *(ELT1 * DCOS( WPTVP1)- COSNP1           )       HITS0636
     1            -CAY2LW *(ELT2 * DCOS( WPTVP2)- COSNP2           )
```

D.10 Subroutine CROSSF (concluded)

```
2 + VCONZ   + DZDT0/ H )                                                HITS0637
      WCON1 = WPTVP1 + PHI1                                             HITS0638
      WCON2 = WPTVP2 + PHI2                                             HITS0639
   IF( IDUMP     .EQ.  1 ) WRITE(6,1000) TP ,ALPHA ,BETA , ATOT         HITS0640
1.VCONY ,VCONZ   ,   YCONA , ZCONA  , VYA , VZA                         HITS0641
     YA   = H *(  CAY1/ ELWP1 * ELT1  * DSIN(WCON1 )-CAY2/ELWP2* ELT2 HITS0642
1 *DSIN(WCON2)- (CAY1LW * DSIN( XNUPH1) -CAY2LW * DSIN(XNUPH2)-DYDTHITS0643
20 /H ) * TP    -CAY1/ ELWP1 * DSIN(XNUPH1+PHI1) +CAY2/ELWP2 * DSIN HITS0644
3(XNUPH2 +PHI2)  + YCONA    )                                           HITS0645
     ZA =  H*(-CAY1/ELWP1*     ELT1   * DCOS(WCON1 )-CAY2/ELWP2* ELT2 HITS0646
1 *DCOS(WCON2)+ (CAY1LW * DCOS(XNUPH1  ) +CAY2LW * DCOS(XNUPH2)+DZDTHITS0647
20 /H ) * TP   + CAY1/ELWP1   * DCOS(XNUPH1+PHI1) +CAY2/ELWP2 * DCOS HITS0648
3(XNUPH2 +PHI2) + ZCONA   )                                             HITS0649
        EOLT =   FEXP( XLAM0 * TP)                                      HITS0650
        EDLT =   DEXP( DELLAM* TP)                                      HITS0651
      ALPMAX =   EOLT *( CAY1 * EDLT   + CAY2 / EDLT )                  HITS0652
      ALPMIN =   EOLT *( CAY1 * EDLT   - CAY2 / EDLT )                  HITS0653
      XJAY = -HV * ( CAY1LW * SINNP1 -CAY2LW * SINNP2   )*1.D3          HITS0654
      XJAZ =   HV * ( CAY1LW * COSNP1 +CAY2LW * COSNP2   )*1.D3          HITS0655
       XJA = DSQRT( XJAY ** 2  + XJAZ ** 2 )                            HITS0656
       IF(IDUMP    .EQ. 1 ) WRITE(6,1001) YA , ZA, EOLT , EDLT          HITS0657
1 , ALPMAX , ALPMIN , XJAY , XJAZ ,  XJA                                HITS0658
1001   FORMAT(  '       YA            ZA            EDLT          EDLT         ALPHITS0659
1MAX         ALPMIN        XJAY          XJAZ          XJA   ' /1X,1P11E12.HITS0660
24 // )                                                                 HITS0661
         RETURN                                                         HITS0662
           END                                                          HITS0663
```

D.11 Subroutine CRVFT

```
      SUBROUTINE CRVFT(KOPT)                                            HITS0664
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0665
      COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS0666
     1IY,KARSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS0667
      COMMON/DEC/DE(600)                                                HITS0668
      DIMENSION DELX(20),FUN(10)                                        HITS0669
C     THIS ROUTINE USES THE T.S. EXPANSIONS OF YP-58,YP3                HITS0670
C     USES 1ST,2ND UNMIXED OR FULL 2ND ORDER FOR KOPT =1,2,3 RESP.      HITS0671
C     CREATE DELX SAMPLE TABLE                                          HITS0672
      DO 1 I=1,K234                                                     HITS0673
      JB=IA(I,2)+1                                                      HITS0674
      IDE=IA(I,1)                                                       HITS0675
    1 DELX(I) = DE(IDE) - B(JB)                                         HITS0676
C     INITIALIZE EXPANSIONS SUMS TO NOMINAL VALUES                      HITS0677
      DO 2 I=1,KD                                                       HITS0678
    2 FUN(I)=C(KC+I-1)                                                  HITS0679
C     START LOOP ON INDEPENDENT UNMIXED DERIVATIVE SUMS                 HITS0680
      DO 4 I=1,K234                                                     HITS0681
      I1 = KC + KD * (2*I-1) - 1                                        HITS0682
      I2=I1+KD                                                          HITS0683
      GO TO(12,11,11),KOPT                                              HITS0684
   12 DO 13 J=1,KD                                                      HITS0685
   13 FUN(J)=FUN(J)+DELX(I)*C(I1+J)                                     HITS0686
      GO TO 4                                                           HITS0687
   11 DO 3 J=1,KD                                                       HITS0688
    3 FUN(J)=FUN(J)+DELX(I)*(C(I1+J)+.5*DELX(I)*C(I2+J))                HITS0689
    4 CONTINUE                                                          HITS0690
C     END OF LOOP --CHECK FOR INCLUSION OF MIXED DERIVATIVES            HITS0691
      IF(KOPT-2)9,9,5                                                   HITS0692
    5 I1=I2+3*KD                                                        HITS0693
      I2=K234-1                                                         HITS0694
      DO 8 I=1,I2                                                       HITS0695
      JB=I+1                                                            HITS0696
      DO 7 J=JB,K234                                                    HITS0697
      DO 6 L=1,KD                                                       HITS0698
    6 FUN(L)=FUN(L)+DELX(I)*DELX(J)*C(I1+L)                             HITS0699
    7 I1=I1+KD                                                          HITS0700
    8 CONTINUE                                                          HITS0701
C     STORE RESULTS IN DE ARRAY                                         HITS0702
    9 DO 10 I=1,KD                                                      HITS0703
      IDE=ID(I,1)                                                       HITS0704
   10 DE(IDE)=FUN(I)                                                    HITS0705
      RETURN                                                            HITS0706
      END                                                               HITS0707
```

D.12 Subroutine DOABC

```
      SUBROUTINE DOABC                                                  HITS0708
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS0709
      LOGICAL*1  BL /' '/                                               HITS0710
      LOGICAL*1  OTT(10) /10H1234567890/                                HITS0711
      LOGICAL*1 FMT1(46)/46H(' IND. VAR.', (A1,' OE(',I4,')=',1PE12.4),/HITS0712
     A/)/                                                               HITS0713
      LOGICAL*1  FMT2(29) /29H(5X,  (A1,' OE(',I4,')  '),/)/            HITS0714
      LOGICAL*1 FMT3(18) /18H(5X,1P  E12.4,///)/                        HITS0715
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,      HITS0716
     1 IOPRNT, IOUT, IY, KARSI(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS0717
      COMMON/OEC/OE(600)                                                HITS0718
      COMMON /CISPRN/ ISPRNT                                            HITS0719
      DIMENSION  IOES(3), JC(3), IOE(2, 600), IB(2,200),                HITS0720
     1 ABCOE(3), OUTOE(10)                                              HITS0721
      EQUIVALENCE  (IOES(1), IAOE), (IOES(2), IBOE), (IOES(3), ICOE),   HITS0722
     1 (JC(1), JCA), (JC(2), JCB), (JC(3), JCC), (OE(1), IOE(1,1)),     HITS0723
     2 (B(1), IB(1,1))                                                  HITS0724
      IF(ISPRNT.EQ.0) WRITE(IOUT,500)                                   HITS0725
  500 FORMAT(1H1,39X,'*** RANGE CHECK INDEPENDENT VARIABLE COMBINATIONS HITS0726
     A***',///)                                                         HITS0727
  501 FORMAT(1H1,39X, '*** RANGE CHECK INDEPENDENT VARIABLE COMBINATION HITS0728
     A***',///)                                                         HITS0729
C SET UP THE FORMATS.                                                   HITS0730
      FMT1(15) = OTT(KT1)                                               HITS0731
      FMT2(6) = OTT(KD)                                                 HITS0732
      FMT3(8) = OTT(KD)                                                 HITS0733
      IF ( KD .LT. 10 )  GO TO 2                                        HITS0734
      FMT2(5) = OTT(1)                                                  HITS0735
      FMT3(7) = OTT(1)                                                  HITS0736
    2 CONTINUE                                                          HITS0737
C IOES(I) IS THE OE SUBSCRIPT FOR THE I.TH TYPE 1 ARRAY.                HITS0738
C JC(I) IS ONE LESS THAN THE ADDRESS IN C FOR THE I.TH TYPE 1 ARRAY.    HITS0739
C THE OE SUBSCRIPTS FOR THE TYPE 1 ARRAYS HAVE BEEN MOVED UP TO THE     HITS0740
C    START OF THE IA ARRAY.                                             HITS0741
C GET THE OE SUBSCRIPTS AND THEIR C ADDRESS.                            HITS0742
      DO 5 I=1,KT1                                                      HITS0743
      IOES(I) = IA(I,1)                                                 HITS0744
      JB = IA(I,2)                                                      HITS0745
    5 JC(I) = IB(1,JB+2) - 1                                            HITS0746
      IAR = 0                                                           HITS0747
   10 IAR = IAR + 1                                                     HITS0748
      IF ( IAOE .GT. 0 )  GO TO 12                                      HITS0749
      L = -IAOE                                                         HITS0750
      IOE(1,L) = C(JCA+IAR)                                             HITS0751
      GO TO 15                                                          HITS0752
   12 OE(IAOE) = C(JCA+IAR)                                             HITS0753
   15 ABCOE(1) = C(JCA+IAR)                                             HITS0754
C IF THERE IS ONLY ONE TYPE 1 ARRAY, IBR=1 TO GET PAST 3 IF.S AT END.   HITS0755
      IBR = 1                                                           HITS0756
      IF ( KT1 .LT. 2 )  GO TO 40                                       HITS0757
```

D.12 Subroutine DOABC (concluded)

```
      GO TO 21                                                          HITS0758
   20 IBR = IBR + 1                                                     HITS0759
   21 IF ( IBOE .GT. 0 )  GO TO 22                                      HITS0760
      L = -IBOE                                                         HITS0761
      IOE(1,L) = C(JCB+IBR)                                             HITS0762
      GO TO 25                                                          HITS0763
   22 OE(IBOE) = C(JCB+IBR)                                             HITS0764
   25 ABCOE(2) = C(JCB+IBR)                                             HITS0765
C IN THE EVENT THERE IS NO THIRD TYPE 1 ARRAY, ICR=1 WILL GET US PAST   HITS0766
C   THE 3 IF,S AT THE END.                                              HITS0767
      ICR = 1                                                           HITS0768
      IF ( KT1  LT, 3 )    GO TO 40                                     HITS0769
      GO TO 31                                                          HITS0770
   30 ICR = ICR + 1                                                     HITS0771
   31 IF ( ICOE .GT. 0 )  GO TO 32                                      HITS0772
      L = -ICOE                                                         HITS0773
      IOE(1,L) = C(JCC+ICR)                                             HITS0774
      GO TO 35                                                          HITS0775
   32 OE(ICOE) = C(JCC+ICR)                                             HITS0776
   35 ABCOE(3) = C(JCC+ICR)                                             HITS0777
   40 CALL PICK1                                                        HITS0778
      IF(ISPRNT.EQ.1) WRITE(IOUT,501)                                   HITS0779
      WRITE (IOUT,FMT1)  (BL, IA(I,1), ABCOE(I), I=1,KT1)               HITS0780
      DO 50 I=1,KD                                                      HITS0781
      JOE = ID(I,1)                                                     HITS0782
   50 OUTOE(I) = OE(JOE)                                                HITS0783
      WRITE (IOUT,FMT2)  (BL, ID(I,1), I=1,KD)                          HITS0784
      WRITE (IOUT,FMT3)  (OUTOE(I), I=1,KD)                             HITS0785
      IF ( ICR .LT. LABC(3) )  GO TO 30                                 HITS0786
      IF ( IBR .LT. LABC(2) )  GO TO 20                                 HITS0787
      IF ( IAR .LT. LABC(1) )  GO TO 10                                 HITS0788
      RETURN                                                            HITS0789
      END                                                               HITS0790
```

D.13 Subroutine DO234

```
      SUBROUTINE DO234                                                   HITS0791
      IMPLICIT REAL*8 (A-H,O-Z)                                          HITS0792
      LOGICAL*1  BL /' '/                                                HITS0793
      LOGICAL*1  OTT(10) /10H1234567890/                                 HITS0794
      LOGICAL*1  FMT2(29) /29H(5X,  (A1,' OE(',I4,')  '),/)/             HITS0795
      LOGICAL*1 FMT3(14) /14H(5X,1P  E12.4)/                             HITS0796
      DIMENSION IBB(2,200)                                               HITS0797
      EQUIVALENCE (IBB(1,1),B(1))                                        HITS0798
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,        HITS0799
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)    HITS0800
      COMMON/OEC/OE(600)                                                 HITS0801
      EQUIVALENCE          (IDUMP   ,OE( 204))                           HITS0802
      DIMENSION  IOES(2), LN(2)                                          HITS0803
      EQUIVALENCE  (IOES(1), IOE1), (IOES(2), IOE2), (LN(1), L1),        HITS0804
     1 (LN(2), L2)                                                       HITS0805
      COMMON/INDEX/JD,I234,ICYBAR,ICSUMS,ICDOUB,KCOV                     HITS0806
      IJB1 = 0                                                           HITS0807
      IJB2 = 0                                                           HITS0808
      IJB3 = 0                                                           HITS0809
      IJB4 = 0                                                           HITS0810
      DO 200 IBOB = 1,KD                                                 HITS0811
      IJB = ID(IBOB,1)                                                   HITS0812
      IF(IJB.EQ.320) IJB1 = IBOB                                         HITS0813
      IF(IJB.EQ.321) IJB2 = IBOB                                         HITS0814
      IF(IJB.EQ.318) IJB3 = IBOB                                         HITS0815
      IF(IJB.EQ.319) IJB4 = IBOB                                         HITS0816
  200 CONTINUE                                                           HITS0817
      IJB = IJB1 + IJB2 + IJB3 + IJB4                                    HITS0818
C SET UP THE FORMATS.                                                    HITS0819
      FMT2(6) = OTT(KD)                                                  HITS0820
      FMT3(8) = OTT(KD)                                                  HITS0821
      IF ( KD .LT. 10 )  GO TO 2                                         HITS0822
      FMT2(5) = OTT(1)                                                   HITS0823
      FMT3(7) = OTT(1)                                                   HITS0824
    2 CONTINUE                                                           HITS0825
C KCYBAR IS ONE LESS THAN THE START OF THE YBAR OUTPUT IN THE C ARRAY.   HITS0826
      KCYBAR = KC                                                        HITS0827
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 201                          HITS0828
      WRITE (IOUT,9003)   (BL, IA(I,1), OE(IA(I,1)), I=1,K234)           HITS0829
 9003 FORMAT('1NOMINAL CASE, INDEPENDENT VARIABLES ...'/// (5X, 5(A1,    HITS0830
     1 '   OE(', I4, ')=', 1PE12.4)))                                    HITS0831
  201 CONTINUE                                                           HITS0832
C THE NOMINAL VALUES.                                                    HITS0833
      CALL PICK1                                                         HITS0834
      CALL STOREC                                                        HITS0835
      K = KC - KD                                                        HITS0836
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 801                          HITS0837
      WRITE (IOUT,9005)                                                  HITS0838
 9005 FORMAT('1NOMINAL CASE, DEPENDENT VARIABLES ...',// )               HITS0839
      WRITE (IOUT,FMT2)   (BL, ID(I,1), I=1,KD)                          HITS0840
```

D.13 Subroutine DO234 (continued)

```
      WRITE (IOUT,FMT3)  (C(K+I), I=1,KD)
C THE FIRST K234 ENTRIES OF THE IA ARRAY ARE THE OE SUBSCRIPTS OF THE
C    INDEPENDENT VARIABLES.
C CHANGE THEM ONE AT A TIME.
      WRITE (IOUT,9006)  (BL, ID(I,1), I=1,KD)
 9006 FORMAT('1LIMITING CASES'// 5X, 'INDEPENDENT VARIABLE', 48X,
     1 'DEPENDENT VARIABLE' // ' NAME      VALUE        TOLERANCE', 5X,
     2 8(A1, ' OE(', I4, ')    ')//35X,2(A1, ' OE(', I4, ')    ') )
  801 CONTINUE
      DO 10 I=1,K234
      IOE = IA(I,1)
      JB = IA(I,2)
      OE(IOE) = B(JB+1) + B(JB+2)
      CALL PICK1
      CALL STOREC
      K = KC - KD
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 202
      WRITE (IOUT,9007)  IOE, B(JB+1), B(JB+2), (C(K+L), L=1,KD)
 9007 FORMAT(1H0,I5,1P2E12.4,4X,8E12.4// 34X,2E12.4,// )
  202 CONTINUE
      OE(IOE) = B(JB+1) - B(JB+2)
      CALL PICK1
      CALL STOREC
      K = KC - KD
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 203
      WRITE (IOUT,9008)  (C(K+L), L=1,KD)
 9008 FORMAT(34X, 1P6E12.4// 34X, 2E12.4,/)
  203 CONTINUE
C RESTORE OE(IOE) TO ITS NOMINAL VALUE.
      OE(IOE) = B(JB+1)
   10 CONTINUE
C IF IOPRNT=1, WE STOP HERE.
      KMBOB = KCYBAR
      IF ( IOPRNT .EQ. 1 )  GO TO 900
C GET THE FIRST AND SECOND UNMIXED PARTIALS AND STORE THEM IN C OVER THE
C    Y(J) PLUS AND Y(J) MINUS ENTRIES.
C KCPAR IS ONE LESS THAN THE STARTING SUBSCRIPT IN THE C ARRAY OF THE
C    PARTIAL DERIVATIVE (Y(J)PLUS - Y(J)MINUS) / (2*DELX(I) ).
      KCPAR = KCYBAR + KD
      KD2 = 2 * KD
      KRP = KCYBAR - KD
      KRM = KCYBAR
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 204
      WRITE (IOUT,9009)  (ID(I,1), I=1,KD)
 9009 FORMAT('1DERIVATIVES' // ' IND. VAR.', 51X, 'DEPENDENT VARIABLE'
     1 // 8X, 10I12 / )
  204 CONTINUE
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $
      IF(IDUMP.EQ.1) GO TO 2000
      DIMENSION IBA(2,200)
```

D.13 Subroutine DO234 (continued)

```
      EQUIVALENCE (IBA(1,1),B(1))                                       HITS0891
      WRITE(IOUT,7000)K234,KCYBAR,KD,KCPAR,KD2,KRP,KRM                  HITS0892
 7000 FORMAT(1H1,7(I5,5X))                                              HITS0893
      WRITE(IOUT,7001)(IA(KARB,1),IA(KARB,2),KARB = 1,200)              HITS0894
 7001 FORMAT(1H ,2(I5,2X))                                              HITS0895
      WRITE(IOUT,7001)(IBA(1,KARB),IBA(2,KARB),KARB = 1,200)            HITS0896
      WRITE(IOUT,7002)(B(KARB),KARB = 1,200)                            HITS0897
 7002 FORMAT(50X,D11.4)                                                 HITS0898
      WRITE(IOUT,7001)(ID(KARB,1),ID(KARB,2),KARB = 1,10)               HITS0899
      WRITE(IOUT,650)(C(KARB),KARB = 1,2000)                            HITS0900
 2000 CONTINUE                                                          HITS0901
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $HITS0902
      DO 30 I=1,K234                                                    HITS0903
      IOE = IA(I,1)                                                     HITS0904
      JB = IA(I,2)                                                      HITS0905
      TWODX = 2.0D0 * B(JB+2)                                           HITS0906
      BSQ = B(JB+2)**2                                                  HITS0907
      KRP = KRP + KD2                                                   HITS0908
      KRM = KRM + KD2                                                   HITS0909
      DO 20 J=1,KD                                                      HITS0910
      KRPJ = KRP + J                                                    HITS0911
      KRMJ = KRM + J                                                    HITS0912
C SAVE YJPLUS.                                                          HITS0913
      YJP = C(KRPJ)                                                     HITS0914
      C(KRPJ) = ( C(KRPJ) - C(KRMJ) ) / TWODX                           HITS0915
   20 C(KRMJ) = -( 2.0D0 * C(KCYBAR+J) - YJP - C(KRMJ) ) /BSQ           HITS0916
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 205                         HITS0917
      WRITE (IOUT,9010)  IOE, (C(KRP+L), L=1,KD)                        HITS0918
 9010 FORMAT(1H0,I6,2X,3H1ST,1P10E12.4)                                 HITS0919
      WRITE (IOUT,9011)   (C(KRM+L), L=1,KD)                            HITS0920
 9011 FORMAT(1H0,8X,3H2ND,1P10E12.4,//)                                 HITS0921
  205 CONTINUE                                                          HITS0922
   30 CONTINUE                                                          HITS0923
      KMBOB = KCYBAR                                                    HITS0924
      IF ( IOPRNT .EQ. 2 )  GO TO 900                                   HITS0925
C FORM THE SUMS.                                                        HITS0926
      KCSUMS = KCPAR + KD * K234 * 2                                    HITS0927
      DO 50 J=1,KD                                                      HITS0928
      KCYBJ = KCYBAR + J                                                HITS0929
      KCSMJ = KCSUMS + J                                                HITS0930
      SYJ = 0.0D0                                                       HITS0931
      SYJS = 0.0D0                                                      HITS0932
      KRP = KCYBAR - KD                                                 HITS0933
      KRM = KCYBAR                                                      HITS0934
      DO 40 I=1,K234                                                    HITS0935
      JB = IA(I,2)                                                      HITS0936
      VAR = B(JB+3)                                                     HITS0937
      KRP = KRP + KD2                                                   HITS0938
      KRM = KRM + KD2                                                   HITS0939
      KRMJ = KRM + J                                                    HITS0940
```

```
      SYJ = SYJ + C(KRMJ) * VAR
      SYJS = SYJS + C(KRP+J)**2 * VAR
   40 CONTINUE
      C(KCSMJ) = C(KCYBJ) + 0.500 * SYJ
      C(KCSMJ+KD) = SYJS
   50 CONTINUE
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 206
C PRINT E(Y(J)) AND E(Y(J)**2).
      WRITE(IOUT,6969)
      WRITE (IOUT,9012)  (ID(I,1), I=1,KD)
 9012 FORMAT(      ' 2ND ORDER MEANS AND 1ST ORDER VARIANCES'/// 62X,
     A'DEPENDENT VARIABLES' // 8X, 10I12)
      WRITE (IOUT,9013)  (C(KCSUMS+I), I=1,KD)
 9013 FORMAT(1H0,4X,5HMEANS,1X,1P10E12.4)
      WRITE (IOUT,9023)  (C(KCSUMS+KD+I), I=1,KD)
 9023 FORMAT(1H0,9HVARIANCES,1X,1P10E12.4)
      DO 7500 I = 1,KD
 7500 CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)
  206 CONTINUE
      JBOB = KCSUMS
      KMBOB = KCYBAR
      IF ( IOPRNT .EQ. 3 )  GO TO 900
C UPDATE KC DUE TO E(Y(J)) AND E(Y(J)**2).
      KC = KC + 2 * KD
      KCDOUB = KC
C VARY THE INDEPENDENT VARIABLES TWO AT A TIME.
      IF((IJB.GT.0),AND,(IDUMP.EQ.0)) GO TO 207
      WRITE (IOUT,9014)  (ID(I,1), I=1,KD)
 9014 FORMAT('1SECOND ORDER MIXED PARTIALS' // ' IND. VAR.', 50X,
     1 'DEPENDENT VARIABLES  '// (8X, 10I12)   )
  207 CONTINUE
C KCDR WILL STORE THE SECOND DERIVATIVES.
      KCDR = KCDOUB - KD
      KM1 = K234 - 1
      KD3 = 3 * KD
      DO 90 I=1,KM1
      IOE = IA(I,1)
      IB = IA(I,2)
      IP1 = I + 1
      DO 80 J=IP1,K234
      JOE = IA(J,1)
      JB = IA(J,2)
      BB = B(IB+2) * B(JB+2)
C THE NEXT DOUBLE DO LOOP WILL CREATE THE FOUR COMBINATIONS OF NOMINAL
C   VALUE PLUS AND MINUS TOLERANCE.
      DO 60 I2=1,2
      OE(IOE) = B(IB+1) + (-1.000)**(3-I2) * B(IB+2) / 2.000
      DO 60 J2 = 1,2
      OE(JOE) = B(JB+1) + (-1.000)**(3-J2) * B(JB+2) / 2.000
      CALL PICK1
```

D.13 Subroutine DO234 (continued)

```
      CALL STOREC
   60 CONTINUE
      KCDR = KCDR + KD
C THE COMBINATIONS OF PLUS AND MINUS ARE KD APART.
      DO 70 L=1,KD
      KCDRJ = KCDR + L
      C(KCDRJ) = ( C(KCDRJ) + C(KCDRJ+KD3) - C(KCDRJ+KD) - C(KCDRJ+
     1 KD2) ) / 88
   70 CONTINUE
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 208
      WRITE (IOUT,9015)  IOE, JOE, (C(KCDR+L), L=1,KD)
 9015 FORMAT('0',2I5, 2X, 1P10E12.4)
  208 CONTINUE
C RESTORE THE SECOND OE OF THE PAIR TO ITS NOMINAL VALUE.
      OE(JOE) = B(JB+1)
C BRING KC UP TO DATE.
      KC = KCDR + KD
   80 CONTINUE
C RESTORE THE FIRST OE OF THE PAIR TO ITS NOMINAL VALUE.
      OE(IOE) = B(IB+1)
   90 CONTINUE
      JBOB = KCSUMS
      KMBOB = KCYBAR
      IF ( IOPRNT .EQ. 4 )  GO TO 900
C
C GAUSSIAN CORRECTION FOR VARIANCES
C
      I234 = K234
      ICYBAR = KCYBAR
      ICSUMS = KCSUMS
      ICDOUB = KCDOUB
      JD = KD
      KCOV = KC
      IF(IOPRNT.EQ.5) GO TO 803
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 805
      WRITE(IOUT,6969)
      WRITE(IOUT,9050)(ID(I,1),I=1,KD)
 9050 FORMAT(' 2ND ORDER MEANS AND VARIANCES ASSUMING GAUSSIAN INDEPENDE
     ANT VARIABLES'///62X, 'DEPENDENT VARIABLES' // 8X, 10I12)
  805 CONTINUE
      CALL GOUC
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 806
      WRITE(IOUT,9013)(C(KCSUMS+I),I = 1,KD)
      WRITE(IOUT,9023)(C(KCSUMS+KD+I),I = 1,KD)
      DO 7501 I = 1,KD
 7501 CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)
  806 CONTINUE
      IF(IOPRNT.EQ.6) GO TO 900
  803 CONTINUE
      KDAVE = (K234 * (K234 + 1))/2
```

D.13 Subroutine DO234 (continued)

```
      DO 8500 IDAV = 1,KDAVE                                            HITS1041
      IDA = KCOV + IDAV                                                 HITS1042
      IF(IDA.GT.5000) WRITE(6,8600) IDA                                 HITS1043
 8600 FORMAT(10X,'IDA = ',I5,1X,'IN DO234 WHICH EXCEEDS THE C ARRAY')  HITS1044
 8500 C(IDA) = 0.0D0                                                    HITS1045
C THE CARD  FOR THIS READ  IS DIRECTLY BEHIND THE NAME-TYPE-NOM-TOL-VAR-HITS1046
C   TABLE LENGTH CARDS.                                                 HITS1047
      WRITE(IOUT,5000)                                                  HITS1048
 5000 FORMAT(1H1,43X,'*** INDEPENDENT VARIABLE CORRELATIONS ***',///,  HITS1049
     A40X,6X,'CODE',6X,6X,'CODE',6X,2X,'CORRELATION',/,                HITS1050
     B40X,5X,'NUMBER',5X,5X,'NUMBER',5X,2X,'COEFFICIENT',///)          HITS1051
  100 READ (INC,9001)  IOE1, IOE2, RHO                                  HITS1052
 9001 FORMAT(I5, I5, F13.1)                                             HITS1053
      IF(IOE1.LT.0) GO TO 802                                           HITS1054
      WRITE(IOUT,5001) IOE1,IOE2,RHO                                    HITS1055
 5001 FORMAT(40X,5X,I5,6X,5X,I5,6X,2X,1PE11.4,//)                       HITS1056
C RUN THRU IA AND GET SUBSCRIPTS (OF THE IA ARRAY) WHERE IOE1 AND      HITS1057
C   IOE2 ARE LOCATED.                                                   HITS1058
      DO 130 J=1,2                                                      HITS1059
      DO 110 I=1,K234                                                   HITS1060
      ISAV = I                                                          HITS1061
      IF ( IA(I,1) .EQ. IOES(J) )  GO TO 120                            HITS1062
  110 CONTINUE                                                          HITS1063
      WRITE (IOUT,9002)                                                 HITS1064
 9002 FORMAT(39H ILLEGAL OE SUBSCRIPT READ IN WITH RHO.)                HITS1065
      CALL EXIT                                                         HITS1066
  120 LN(J) = ISAV                                                      HITS1067
  130 CONTINUE                                                          HITS1068
C GET THE SMALLEST IA SUBSCRIPT INTO LN(1).                             HITS1069
      IF ( LN(1) .LT. LN(2) )  GO TO 140                                HITS1070
      LS = LN(1)                                                        HITS1071
      LN(1) = LN(2)                                                     HITS1072
      LN(2) = LS                                                        HITS1073
C GET THE SUBSCRIPTS TO PICK UP THE VARIANCES IN THE B ARRAY.           HITS1074
  140 IB1 = IA(L1,2)                                                    HITS1075
      IB2 = IA(L2,2)                                                    HITS1076
      COV = RHO * DSQRT( B(IB1+3) * B(IB2+3) )                          HITS1077
C                                                                       HITS1078
C  STORE COVARIANCES IN THE C ARRAY                                     HITS1079
C                                                                       HITS1080
      MCOV = NCOV(L1,L2)                                                HITS1081
      C(MCOV) = COV                                                     HITS1082
      GO TO 100                                                         HITS1083
  802 CONTINUE                                                          HITS1084
C                                                                       HITS1085
C  COMPUTE 2ND ORDER MEANS AND 1ST ORDER VARIANCES WITH CORRELATIONS    HITS1086
      CALL QCOR                                                         HITS1087
      IF(IOPRNT.EQ.7) GO TO 804                                         HITS1088
      IF((IJ3.GT.0).AND.(IDUMP.EQ.0)) GO TO 807                         HITS1089
      WRITE(IOUT,6960)                                                  HITS1090
```

```
      WRITE(IOUT,9051) (ID(I,1),I = 1,KD)                               HITS1091
 9051 FORMAT(      ' 2ND ORDER MEANS AND 1ST ORDER VARIANCES WITH CORRELAHITS1092
     ATED INDEPENDENT VARIABLES'///62X, 'DEPENDENT VARIABLES' //        HITS1093
     88X,10I12)                                                          HITS1094
      WRITE(IOUT,9013)(C(KCSUMS+I),I = 1,KD)                            HITS1095
      WRITE(IOUT,9023)(C(KCSUMS+KD+I),I = 1,KD)                         HITS1096
      DO 7502 I = 1,KD                                                  HITS1097
 7502 CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)                             HITS1098
 807  CONTINUE                                                          HITS1099
      GO TO 900                                                         HITS1100
C                                                                       HITS1101
C   GAUSSIAN CORRELATED INPUTS                                          HITS1102
C                                                                       HITS1103
 804  CONTINUE                                                          HITS1104
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 825                         HITS1105
      WRITE(IOUT,6969)                                                  HITS1105
      WRITE(IOUT,9052) (ID(I,1),I=1,KD)                                 HITS1107
 9052 FORMAT(' 2ND ORDER MEANS AND VARIANCES ASSUMING CORRELATED GAUSSIAHITS1108
     AN INDEPENDENT VARIABLES'///62X, 'DEPENDENT VARIABLES' //          HITS1109
     88X,10I12)                                                          HITS1110
 825  CONTINUE                                                          HITS1111
      CALL GQCOR                                                        HITS1112
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 900                         HITS1113
      WRITE(IOUT,9013)(C(KCSUMS+I),I = 1,KD)                            HITS1114
      WRITE(IOUT,9023)(C(KCSUMS+KD+I),I=1,KD)                           HITS1115
      DO 7503 I = 1,KD                                                  HITS1116
 7503 CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)                             HITS1117
 900  CONTINUE                                                          HITS1118
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $HITS1119
      IF(IDUMP.EQ.0) GO TO 2001                                         HITS1120
      WRITE(IOUT,650)(C(KIJ),KIJ = 1,2000)                              HITS1121
 2001 CONTINUE                                                          HITS1122
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $HITS1123
      IF(IJB.EQ.0) RETURN                                               HITS1124
      IF(IJB1.EQ.0) GO TO 225                                           HITS1125
      MBOB = 0                                                          HITS1126
      IJBOB = IJB1                                                      HITS1127
      WRITE(IOUT,8000)                                                  HITS1128
 8000 FORMAT(1H1,50X,14H*** NOTICE ***,//,                              HITS1129
     A30X,'      THE SPECIFICATION OF EITHER RHOT OR RHOX (IE        ',/,HITS1130
     B30X,'VARIABLES 320 OR 321) OR BOTH AS OUTPUT VARIABLES          ',/,HITS1131
     C30X,'NECESSITATES UP-DATING THE NOMINAL VALUES, DERIVATIVES,  ',/,HITS1132
     D30X,'MEANS, AND VARIANCES. THE LIMITING CASES FOR AND PARTIAL ',/,HITS1133
     E30X,'DERIVATIVES OF RHOT AND RHOX ARE MEANINGLESS. THUS THE   ',/,HITS1134
     F30X,'FULL PROJECTILE DISPERSION CODE SHOULD BE UTILIZED IN    ',/,HITS1135
     G30X,'MONTE CARLO SIMULATIONS (MCALC = 0) TO OBTAIN            ',/,HITS1136
     H30X,'MEANINGFULL VALUES OF RHOT AND/OR RHOX. IF A TAYLOR      ',/,HITS1137
     I30X,'SERIES APPROXIMATION OF THE PROJECTILE DISPERSION CODE   ',/,HITS1138
     J30X,'(MCALC = 1, 2, OR 3) IS USED IN MONTE CARLO              ',/,HITS1139
     K30X,'SIMULATIONS THE RHOT AND RHOX RESULTS ARE MEANINGLESS.   ',  HITS1140
```

D.13 Subroutine DO234 (continued)

```
      DO 503  IBOB = 1,KD                                               HITS1141
      IF(ID(IBOB,1).EQ.306) GO TO 504                                   HITS1142
      IF(ID(IBOB,1).EQ.307) GO TO 505                                   HITS1143
      IF(ID(IBOB,1).EQ.308) GO TO 506                                   HITS1144
      GO TO 503                                                         HITS1145
  504 DXTO = C(KMBOB + IBOB)                                            HITS1146
      MBOB = MBOB + 1                                                   HITS1147
      IF (IOPRNT.LT.3) GO TO 503                                        HITS1148
      OE(301) = C(JBOB + IBOB)                                          HITS1149
      BVAR1 = C(JBOB + KD + IBOB)                                       HITS1150
      GO TO 503                                                         HITS1151
  505 DYTO = C(KMBOB + IBOB)                                            HITS1152
      MBOB = MBOB + 1                                                   HITS1153
      IF(IOPRNT.LT.3) GO TO 503                                         HITS1154
      OE(302) = C(JBOB + IBOB)                                          HITS1155
      BVAR2 = C(JBOB + KD + IBOB)                                       HITS1156
      GO TO 503                                                         HITS1157
  506 DZTO = C(KMBOB + IBOB)                                            HITS1158
      MBOB = MBOB + 1                                                   HITS1159
      IF(IOPRNT.LT.3) GO TO 503                                         HITS1160
      OE(303) = C(JBOB + IBOB)                                          HITS1161
      BVAR3 = C(JBOB + KD + IBOB)                                       HITS1162
  503 CONTINUE                                                          HITS1163
      IF(MBOB.EQ.3) GO TO 507                                           HITS1164
      WRITE(IOUT,601)                                                   HITS1165
  601 FORMAT(1H1,///,80HVARIABLES DXT, DYT, DZT (IE 306, 307, 308) MUST HITS1166
     ABE DEFINED AS EITHER TYPE 7 OR 8,//,1H1)                          HITS1167
      CALL EXIT                                                         HITS1168
  507 C(KMBOB+IJBOB)=DSQRT((DXTO-OE(301))**2+(DYTO-OE(302))**2+         HITS1169
     A(DZTO-OE(303))**2)                                                HITS1170
      DO 221 IBOB = 1,K234                                              HITS1171
      C(KMBOB+IJBOB+KD+2*(IBOB-1)*KD) = 0.0D0                           HITS1172
  221 C(KMBOB+IJBOB+2*IBOB*KD)=0.0D0                                    HITS1173
      IF(IOPRNT.LT.3) GO TO 220                                         HITS1174
  508 AVAR = (DSQRT(BVAR1) + DSQRT(BVAR2) + DSQRT(BVAR3))/3.            HITS1175
      C(JBOB + IJBOB) = 1.5957690D0 * AVAR                              HITS1175
      C(JBOB + IJBOB + KD) = 0.4535209D0 * AVAR * AVAR                  HITS1177
      IF(IOPRNT.LT.4) GO TO 220                                         HITS1178
      KAR = (K234-1) * K234/2                                           HITS1179
      DO 222 IBOB = 1,KAR                                               HITS1180
  222 C(KCDOUB+IJBOB+(IBOB-1)*KD) = 0.0D0                               HITS1181
  220 CONTINUE                                                          HITS1182
      IJB = IJB - IJB1                                                  HITS1183
      IF(IJB.EQ.0) GO TO 299                                            HITS1184
  225 CONTINUE                                                          HITS1185
      IF (IJB2.EQ.0) GO TO 226                                          HITS1186
      MBOB = 0                                                          HITS1187
      IJBOB = IJB2                                                      HITS1188
      IF(IJB1.EQ.0) WRITE(IOUT,8000)                                    HITS1189
      DO 703 IBOB = 1,KD                                                HITS1190
```

D.13 Subroutine DO234 (continued)

```
      IF(ID(IBOB,1).EQ.309) GO TO 704
      IF(ID(IBOB,1).EQ.310) GO TO 705
      GO TO 703
  704 DYXO = C(KMBOB+IBOB)
      MBOB = MBOB + 1
      IF(IOPRNT.LT.3) GO TO 703
      OE(304) = C(JBOB + IBOB)
      BVAR1 = C(JBOB + IBOB + KD)
      GO TO 703
  705 DZXO = C(KMBOB+IBOB)
      MBOB = MBOB + 1
      IF (IOPRNT.LT.3) GO TO 703
      OE(305) = C(JBOB + IBOB)
      BVAR2 = C(JBOB + IBOB + KD)
  703 CONTINUE
      IF(MBOB.EQ.2) GO TO 707
      WRITE(IOUT,610)
  610 FORMAT(1H1,////,74HVARIABLES DYX AND DZX (IE 309, 310) MUST BE DEF
     AINED AS EITHER TYPES 7 OR 8,///,1H1)
      CALL EXIT
  707 C(KMBOB+IJBOB)=DSQRT((DYXO-OE(304))**2+(DZXO-OE(305))**2)
      DO 230 IBOB = 1, K234
      C(KMBOB+IJBOB+KD+2*(IBOB-1)*KD) = 0.0D0
  230 C(KMBOB+IJBOB+2*IBOB*KD)=0.0D0
      IF (IOPRNT.LT.3) GO TO 240
  708 AVAR = (DSQRT(BVAR1) + DSQRT(BVAR2))/2.
      C(JBOB + IJBOB) = 1.2533141D0 * AVAR
      C(JBOB + IJBOB + KD) = 0.42920367D0 * AVAR * AVAR
      IF (IOPRNT.LT.4) GO TO 240
      KAR = (K234-1) * K234/2
      DO 232 IBOB = 1,KAR
  232 C(KCDOUB+IJBOB+(IBOB-1)*KD) = 0.0D0
  240 CONTINUE
      IJB = IJB - IJB2
      IF (IJB.EQ.0) GO TO 199
  226 CONTINUE
      WRITE(IOUT,8001)
 8001 FORMAT(1H1,50X,14H*** NOTICE ***,//,
     A30X,'       THE SPECIFICATION OF EITHER RADT OR RADX (IE ',/,
     B30X,'VARIABLES 318 OR 319) OR BOTH AS OUTPUT VARIABLES ',/,
     C30X,'NECESSITATES UP-DATING THE DERIVATIVES, MEANS,  ',/,
     D30X,'AND VARIANCES. PARTIAL DERIVATIVES OF RADT AND RADX ARE  ',/,
     E30X,'MEANINGLESS. THUS THE FULL PROJECTILE DISPERSION CODE ',/,
     F30X,'SHOULD BE UTILIZED IN MONTE CARLO SIMULATIONS (MCALC = 0)',/,
     G30X,'TO OBTAIN MEANINGFULL VALUES OF RADT AND/OR RADX. IF     ',/,
     H30X,'A TAYLOR SERIES APPROXIMATION OF THE PROJECTILE ',/,
     I30X,'DISPERSION CODE (MCALC = 1, 2, OR 3) IS USED IN MONTE ',/,
     J30X,'CARLO SIMULATIONS THE RADT AND RADX RESULTS ARE ',/,
     K30X,'MEANINGLESS. RADT AND/OR RADX MEANS AND VARIANCES ARE ',/,
     L30X,'TAKEN FROM THE TYPE 8 SPECIFICATION.')
```

```
      IF (IJB3.EQ.0) GO TO 300
      IJBOB = IJB3
      IF ( IOPRNT.EQ.1) GO TO 350
      DO 351 IBOB = 1,K234
      C(KMBOB+IJBOB+KD+2*(IBOB-1)*KD) = 0.0D0
  351 C(KMBOB+IJBOB+2*IBOB*KD) = 0.0D0
      IF (IOPRNT.EQ.2) GO TO 350
      IF (        ID(IJBOB,2) .NE.0) GO TO 301
      WRITE(IOUT,8002)
 8002 FORMAT(1H1,///,38H VARIABLE RADT (IE 318) MUST BE TYPE 8.///,1H1)
      CALL EXIT
  301 C(JBOB+IJBOB) = B(ID(IJBOB,2)+1)
      C(JBOB+IJBOB+KD) = B(ID(IJBOB,2)+3)
      IF (IOPRNT.EQ.3) GO TO 350
      KAR = (K234-1)*K234/2
      DO 302  IBOB = 1,KAR
  302 C(KCDOUB+IJBOB+(IBOB-1)*KD) = 0.0D0
  350 CONTINUE
      IF ( IJB4.EQ.0) GO TO 299
  300 CONTINUE
      IJBOB = IJB4
      IF (IOPRNT.EQ.1) GO TO 299
      DO 361 IBOB = 1,K234
      C(KMBOB+IJBOB+KD+2*(IBOB-1)*KD) = 0.0D0
  361 C(KMBOB+IJBOB+2*IBOB*KD) = 0.0D0
      IF (IOPRNT.EQ.2) GO TO 299
      IF (        ID(IJBOB,2) .NE.0) GO TO 362
      WRITE(IOUT,8003)
 8003 FORMAT(1H1,///,38H VARIABLE RADX (IE 319) MUST BE TYPE 8.//,1H1)
      CALL EXIT
  362 C(JBOB+IJBOB) = B(ID(IJBOB,2)+1)
      C(JBOB+IJBOB+KD) = B(ID(IJBOB,2)+3)
      IF(IOPRNT.EQ.3) GO TO 299
      KAR = (K234-1)*K234/2
      DO 369 IBOB = 1,KAR
  369 C(KCDOUB+IJBOB+(IBOB-1)*KD) = 0.0D0
  299 CONTINUE
      WRITE(IOUT,9003) (BL,IA(I,1),B(IA(I,2)+1),I = 1,K234)
      WRITE(IOUT,9005)
      WRITE(IOUT,FMT2) (BL,ID(I,1), I = 1,KD)
      WRITE(IOUT,FMT3) (C(KMBOB+I), I = 1,KD)
      IF (IOPRNT.GT.1) GO TO 380
      WRITE(IOUT,9006) (BL,ID(I,1), I = 1,KD)
      DO 381 I = 1,K234
      WRITE(IOUT,9007) IA(I,1),B(IA(I,2)+1),B(IA(I,2)+2),
     A(C(KMBOB+2*KD*(I-1)+KD+L), L = 1,KD)
  381 WRITE(IOUT,9008) (C(KMBOB+2*KD*I+L), L = 1,KD)
      RETURN
  380 WRITE(IOUT,9009) (ID(I,1), I = 1,KD)
      DO 382 I = 1,K234
```

D.13 Subroutine DO234 (concluded)

```
      WRITE(IOUT,9010) IA(I,1),(C(KMBOB+KD+L+2*KD*(I-1)), L = 1,KD)     HITS1291
  382 WRITE(IOUT,9011)(C(KMBOB+2*KD*I+L), L = 1,KD)                     HITS1292
      IF (IOPRNT.LT.3) RETURN                                           HITS1293
      IF (IOPRNT.EQ.3) GO TO 385                                        HITS1294
      WRITE(IOUT,9014) (ID(I,1),I = 1,KD)                               HITS1295
      KMM = K234 - 1                                                    HITS1296
      LMNO = KCDOUB                                                     HITS1297
      DO 383 I = 1 , KMM                                                HITS1298
      KMMM = I + 1                                                      HITS1299
      DO 384 J = KMMM,K234                                              HITS1300
      WRITE(IOUT,9015) IA(I,1),IA(J,1),(C(LMNO+L),L = 1,KD)             HITS1301
  384 LMNO = LMNO + KD                                                  HITS1302
  383 CONTINUE                                                          HITS1303
  385 WRITE(IOUT,6969)                                                  HITS1304
 6969 FORMAT(1H1)                                                       HITS1305
      KPRNT = IOPRNT - 2                                                HITS1306
      GO TO (875,875,875,877,878),KPRNT                                 HITS1307
  875 WRITE(IOUT,9012)(ID(I,1),I = 1,KD)                                HITS1308
      GO TO 880                                                         HITS1309
  876 WRITE(IOUT,9051)(ID(I,1),I = 1,KD)                                HITS1310
      GO TO 880                                                         HITS1311
  877 WRITE(IOUT,9050)(ID(I,1),I = 1,KD)                                HITS1312
      GO TO 880                                                         HITS1313
  878 WRITE(IOUT,9052)(ID(I,1),I = 1,KD)                                HITS1314
  880 CONTINUE                                                          HITS1315
      WRITE(IOUT,9013)(C(JBOB+I), I = 1,KD)                             HITS1316
      WRITE(IOUT,9023) (C(JBOB+KD+I), I = 1,KD)                         HITS1317
      DO 7504 I = 1,KD                                                  HITS1318
 7504 CALL STDDEV(C(JBOB   +KD+I),I,KD,IOUT)                            HITS1319
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $HITS1320
      IF(IDUMP.EQ.0) GO TO 2002                                         HITS1321
      WRITE(IOUT,650)(C(KIJ),KIJ = 1,2000)                              HITS1322
  650 FORMAT(' C = ',1P10E12.4)                                         HITS1323
 2002 CONTINUE                                                          HITS1324
C$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $HITS1325
      RETURN                                                            HITS1326
      END                                                               HITS1327
```

D.14 Subroutine EXTRA

```
      SUBROUTINE EXTRA                                                  HITS1328
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1329
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,       HITS1330
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS1331
      DIMENSION IB(2,200)                                               HITS1332
      EQUIVALENCE (IB(1,1),B(1))                                        HITS1333
      IF(KT1.EQ.0) GO TO 100                                            HITS1334
C                                                                       HITS1335
C   PRINT RANGE CHECK VALUES                                            HITS1336
      WRITE(IOUT,5000)                                                  HITS1337
      DO 1 I = 1,KT1                                                    HITS1338
      IOE = IA(I,1)                                                     HITS1339
      KB = IA(I,2) + 1                                                  HITS1340
      ITL = IB(1,KB)                                                    HITS1341
      KC = IB(1,KB+1)                                                   HITS1342
 1    WRITE(IOUT,5001) IOE,(C(KC-1+J),J=1,ITL)                          HITS1343
 5000 FORMAT(1H1,49X,'*** RANGE CHECK VALUES ***',///,                  HITS1344
     A22X,6X,'CODE',6X,7X,'1',8X,7X,'2',8X,7X,'3',8X,7X,'4',8X,7X,'5',/,HITS1345
     B22X,5X,'NUMBER',5X,5(5X,'VALUE',6X),//)                           HITS1346
 5001 FORMAT(22X,4X,I5,6X,5(3X,1PE11.4,2X),/,5(26X,5(3X,1PE11.4,2X),/),/HITS1347
     A/)                                                                HITS1348
 100  CONTINUE                                                          HITS1349
      IF(K234.EQ.0) RETURN                                              HITS1350
      DO 2 I = 1, K234                                                  HITS1351
      IOE = IA(I,1)                                                     HITS1352
      KB = IA(I,2)                                                      HITS1353
      IF(IB(1,KB).NE.4) GO TO 2                                         HITS1354
      WRITE(IOUT,5003) IOE                                              HITS1355
 5003 FORMAT(1H1, 38X,'*** CODE NUMBER',    I5,1X,                      HITS1356
     1                                      'PROBABILITY DENSITY FUNCTIHITS1357
     2ON ***'                                                           HITS1358
     A    ,///,41X,5X,'POINT',6X,3X,'COORDINATE',3X,2X,'PROBABILITY',3X,/HITS1359
     B,41X,5X,'NUMBER',5X,5X,'VALUE',6X,4X,'DENSITY',///)               HITS1360
      ITL = IB(1,KB+4)                                                  HITS1361
      KC = IB(1,KB+5)                                                   HITS1362
      DO 3 J = 1,ITL                                                    HITS1363
 3    WRITE(IOUT,5004) J,C(KC-1+ITL+J),C(KC-1+J)                        HITS1364
 2    CONTINUE                                                          HITS1365
 5004 FORMAT(40X,6X,I4,6X,2(3X,1PE11.4,2X))                             HITS1366
      RETURN                                                            HITS1367
      END                                                               HITS1368
```

D.15 Subroutine FC2987

```
      SUBROUTINE FC2987                                                 HITS1369
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1370
      COMMON/OEC/OE(600)                                                HITS1371
      EQUIVALENCE             (VO      ,OE(   1)), (VWIND  ,OE(   2)),  HITS1372
     A(PHIWIN ,OE(   3)), (RHO     ,OE(   4)), (CD1    ,OE(   5)),     HITS1373
     B(V1     ,OE(   6)), (CD2     ,OE(   7)), (V2     ,OE(   8)),     HITS1374
     C(XN     ,OE(   9)), (CNA     ,OE(  10)), (SM     ,OE(  11)),     HITS1375
     D(CMQ    ,OE(  12)), (CMPA    ,OE(  13)), (TRIM   ,OE(  14)),     HITS1376
     E(PNITRI ,OE(  15)), (ANGO    ,OE(  16)), (PHIANG ,OE(  17)),     HITS1377
     F(RATE   ,OE(  18)), (PHIRAT  ,OE(  19)), (GAMO   ,OE(  20)),     HITS1378
     G(PHIGAM ,OE(  21)), (A       ,OE(  22)), (ELL    ,OE(  23)),     HITS1379
     H(D      ,OE(  24)), (W       ,OE(  25)), (AIX    ,OE(  26)),     HITS1380
     I(AIY    ,OE(  27)), (P       ,OE(  28)), (ALPCON ,OE(  29)),     HITS1381
     J(CAA    ,OE(  30))                                                HITS1382
      EQUIVALENCE             (ALPHA   ,OE( 400)), (ATOT   ,OE( 401)),  HITS1383
     A(ALPTRM ,OE( 402)), (ALPHDO  ,OE( 403)), (B      ,OE( 404)),     HITS1384
     B(BETTRM ,OE( 405)), (BETAO   ,OE( 406)), (BETADO ,OE( 407)),     HITS1385
     C(BETA   ,OE( 408)), (ALPMAX  ,OE( 409)), (ALPMIN ,OE( 410)),     HITS1386
     D(CAYD   ,OE( 411)), (CD8     ,OE( 412)), (CMA    ,OE( 413)),     HITS1387
     E(CMTHTD ,OE( 414)), (CMTHTA  ,OE( 415)), (CAY1   ,OE( 416)),     HITS1388
     F(CAY2   ,OE( 417)), (CDAOB   ,OE( 418)), (CDAB   ,OE( 419)),     HITS1389
     G(DELTT  ,OE( 420)), (DELV    ,OE( 421)), (DELT   ,OE( 422)),     HITS1390
     H(DELX   ,OE( 423)), (DYDXO   ,OE( 424)), (DZDXO  ,OE( 425)),     HITS1391
     I(DYDT   ,OE( 426)), (DZDTO   ,OE( 427)), (DELW   ,OE( 428)),     HITS1392
     J(DELLA  ,OE( 429)), (EYEP    ,OE( 430)), (EMP    ,OE( 431)),     HITS1393
     K(EOLT   ,OE( 432)), (EDLT    ,OE( 433)), (F      ,OE( 434)),     HITS1394
     L(H      ,OE( 435)), (PSIDO   ,OE( 436)), (PHIO   ,OE( 437)),     HITS1395
     M(PHI1   ,OE( 438)), (PHI2    ,OE( 439)), (PSIO   ,OE( 440)),     HITS1396
     N(R1     ,OE( 441)), (R2      ,OE( 442)), (R3     ,OE( 443)),     HITS1397
     O(R4     ,OE( 444)), (RTRIM   ,OE( 445)), (TC     ,OE( 446)),     HITS1398
     P(TCO    ,OE( 447)), (TS      ,OE( 448)), (TGO    ,OE( 449)),     HITS1399
     Q(TP     ,OE( 450)), (THETAO  ,OE( 451)), (THETDO ,OE( 452)),     HITS1400
     R(TOL    ,OE( 453)), (VC      ,OE( 454)), (VCO    ,OE( 455)),     HITS1401
     S(VYA    ,OE( 456)), (VZA     ,OE( 457)), (WX     ,OE( 458))      HITS1402
      EQUIVALENCE             (WZ      ,OE( 459)), (WO     ,OE( 460)),  HITS1403
     A(W1     ,OE( 461)), (W2      ,OE( 462)), (WL2    ,OE( 463)),     HITS1404
     B(XLAM   ,OE( 464)), (XLAM1   ,OE( 465)), (XLAM2  ,OE( 466)),     HITS1405
     C(XNU1   ,OE( 467)), (XNU2    ,OE( 468)), (XG     ,OE( 469)),     HITS1406
     D(XMU    ,OE( 470)), (XJAY    ,OE( 471)), (XJAZ   ,OE( 472)),     HITS1407
     E(XJA    ,OE( 473)), (YA      ,OE( 474)), (ZA     ,OE( 475)),     HITS1408
     F(ALPH   ,OE( 476))                                                HITS1409
      EQUIVALENCE             (TN      ,OE( 500)), (XTFC   ,OE( 501)),  HITS1410
     A(YTFC   ,OE( 502)), (ZTFC    ,OE( 503)), (VXTFC  ,OE( 504)),     HITS1411
     B(VYTFC  ,OE( 505)), (VZTFC   ,OE( 506))                          HITS1412
      CALL INCONF                                                       HITS1413
      A57 = ALPCON/57.29577951D0                                        HITS1414
      CALL XXC(XLAMO,XC,VO,DELLAM,CAY1,CAY2,TOL,A57)                    HITS1415
      CALL TRAJXF(XN,XC,VX,TN)                                          HITS1416
      CALL CROSSF(XN,VX)                                                HITS1417
      CALL WIND(VX,TM,XN,WZ,WX,VO,VZW,ZW)                               HITS1418
```

D.15 Subroutine FC2987 (concluded)

```
      XTFC  = XN                                                        HITS1419
      YTFC  = YA                                                        HITS1420
      ZTFC  = ZA + ZW                                                   HITS1421
      VXTFC = VX                                                        HITS1422
      VYTFC = VYA                                                       HITS1423
      VZTFC = VZA + VZW                                                 HITS1424
      RETURN                                                            HITS1425
      END                                                               HITS1426
```

D.16 Function FEXP

```
      FUNCTION FEXP(X)                                                  HITS1427
      IMPLICIT REAL*8(A-H, O-Z)                                         HITS1428
      IF( X .LT. -90.D0)  GO TO 1                                       HITS1429
      FEXP= DEXP( X)                                                    HITS1430
      RETURN                                                            HITS1431
    1 FEXP= 0.D0                                                        HITS1432
      RETURN                                                            HITS1433
      END                                                               HITS1434
```

D.17 Subroutine FILLA

```
      SUBROUTINE FILLA(A,FCTR,PM,PT,CM,CV,NCELL)                       HITS1435
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1436
      DIMENSION A(20,5)                                                 HITS1437
C     THIS ROUTINE FILLS OUT THE OUTPUT ARRAY                           HITS1438
      CELL=2.D0*PT/FCTR                                                 HITS1439
      ENN=0.D0                                                          HITS1440
      SUM1=0.D0                                                         HITS1441
      SUM2=0.D0                                                         HITS1442
C     COUNT TOTAL FREQUENCY                                             HITS1443
      DO 1 I=1,NCELL                                                    HITS1444
    1 ENN=ENN+A(I,4)                                                    HITS1445
      DO 2 I=1,NCELL                                                    HITS1446
      A(I,5)=A(I, 4)/ENN                                                HITS1447
      FI=I                                                              HITS1448
      FIM=FI-.5D0                                                       HITS1449
      FI=FIM*A(I,5)                                                     HITS1450
      SUM1=SUM1+FI                                                      HITS1451
      SUM2=SUM2+FI*FIM                                                  HITS1452
      A(I,5) = A(I,5) / FCTR                                            HITS1453
      A(I,1)=PM+PT*(2.*FIM/CELL-1.)                                     HITS1454
      A(I,2)=A(I,1)-.5D0*FCTR                                           HITS1455
    2 A(I,3)=A(I,2)+FCTR                                                HITS1456
      CV=(SUM2-SUM1**2)*FCTR**2                                         HITS1457
      CM=PM+FCTR*(SUM1-.5D0*CELL)                                       HITS1458
      RETURN                                                            HITS1459
      END                                                               HITS1460
```

D.18 Subroutine FILLIN

```
      SUBROUTINE FILLIN(IA ,IOUT)                                      HITS1461
      IMPLICIT REAL*8 (A-H,O-Z)                                        HITS1462
      COMMON/NAMES/HEAD(180)                                           HITS1463
      IAT = IA                                                         HITS1464
      IF (IA.LT.0) IAT = - IA                                          HITS1465
      IF(IAT.GT.100) GO TO 1                                           HITS1466
      WRITE(IOUT,8000)         HEAD(IAT),IAT                           HITS1467
      RETURN                                                           HITS1468
 1    CONTINUE                                                         HITS1469
      IF(IAT.GT.200) GO TO 2                                           HITS1470
      WRITE(IOUT,8001)         HEAD(IAT-100),IAT                       HITS1471
      RETURN                                                           HITS1472
 2    CONTINUE                                                         HITS1473
      IF(IAT.GT.300) GO TO 3                                           HITS1474
      WRITE(IOUT,8002)         HEAD(IAT-165),IAT                       HITS1475
      RETURN                                                           HITS1476
 3    CONTINUE                                                         HITS1477
      IF(IAT.GE.400) GO TO 4                                           HITS1478
      WRITE(IOUT,8003) HEAD(IAT-259), IAT                              HITS1479
      RETURN                                                           HITS1480
 4    CONTINUE                                                         HITS1481
      IF(IAT.GE.500) GO TO 5                                           HITS1482
      WRITE(IOUT,8004) HEAD(IAT-315), IAT                              HITS1483
      RETURN                                                           HITS1484
 5    CONTINUE                                                         HITS1485
      IF(IAT.GT.600) GO TO 6                                           HITS1486
      WRITE(IOUT,8004) HEAD(IAT-338), IAT                              HITS1487
      RETURN                                                           HITS1488
 6    WRITE(IOUT,8005)                                                 HITS1489
      RETURN                                                           HITS1490
8000 FORMAT(1H ,36X,A6,1X,'FIRE CONTROL   ',9X,' CODE NUMBER ',I3)     HITS1491
8001 FORMAT(1H ,36X,A6,1X,'REAL WORLD     ',9X,' CODE NUMBER ',I3)     HITS1492
8002 FORMAT(1H ,36X,A6,1X,'SYSTEM CONTROL',9X,' CODE NUMBER ',I3)      HITS1493
8003 FORMAT(1H ,36X,A6,1X,'STATISTICAL VARIABLE',3X,' CODE NUMBER ',I3)HITS1424
8004 FORMAT(1H ,36X,A6,1X,'TRAJECTORY VARIABLE ',3X,' CODE NUMBER ',I3)HITS1495
8005 FORMAT(1H1,50(/,1X,'OE ADDRESS EXCEEDS 600'))                     HITS1496
      END                                                              HITS1497
```

D.19 Subroutine GQCOR

```
      SUBROUTINE    GQCOR                                           HITS1498
      IMPLICIT REAL*8(A-H, O-Z)                                     HITS1499
      COMMON /CARDCM/  B(200), C(5000), IA(200,2)                   HITS1500
      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV         HITS1501
      IF( K234 .EQ. 1) RETURN                                       HITS1502
      K1 = K234 -1                                                  HITS1503
      DO  1  I = 1, KD                                              HITS1504
      DELPP = 0.D0                                                  HITS1505
      DO  2   J =  1 , K234                                         HITS1506
      NV =  NVARX( J)                                               HITS1507
      VARXJ =  B(NV)                                                HITS1508
      DO  2   L =  1 , K1                                           HITS1509
      N2L =   N2D(I,L , J )                                         HITS1510
      DYXLJ =  C( N2L)                                              HITS1511
      L1 = L + 1                                                    HITS1512
      DO  2  K =  L1 , K234                                         HITS1513
      N2K = N2D(I,K,J)                                              HITS1514
      NC = NCOV(L,K)                                                HITS1515
    2 DELPP  =  DELPP  + DYXLJ * C(N2K) *  C(NC) * VARXJ            HITS1516
      DO  3  J =  1, K1                                             HITS1517
      J1 = J+1                                                      HITS1518
      DO  3  M = J1 , K234                                          HITS1519
      NC =  NCOV(J,M)                                               HITS1520
      COVJM =  C(NC)                                                HITS1521
      DO  3  L = 1, K1                                              HITS1522
      N2 =  N2D(I,L, J)                                             HITS1523
      DYLJ =  C(N2)                                                 HITS1524
      L1 = L+ 1                                                     HITS1525
      DO 3  K  =  L1 , K234                                         HITS1526
      N2 = N2D(I,K,M )                                              HITS1527
      NC = NCOV(L,K)                                                HITS1528
    3 DELPP = DELPP+ DYLJ * C(N2) * C(NC) * COVJM                   HITS1529
      DELPP = 2.D0 * DELPP                                          HITS1530
      NY = NVAR(I)                                                  HITS1531
    1 C(NY) = C(NY)  +  DELPP                                       HITS1532
      RETURN                                                        HITS1533
      END                                                           HITS1534
```

D.20 Subroutine GQUC

```
          SUBROUTINE   GQUC                                               HITS1535
      IMPLICIT REAL*8(A-H, O-Z)                                           HITS1536
      COMMON /CARDCM/  B(200), C(5000), IA(200,2)                         HITS1537
       COMMON  /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV             HITS1538
            KD = K234 - 1                                                 HITS1539
          DO  1  I =  1 , KD                                              HITS1540
           SUM = 0.D0                                                     HITS1541
          DO  2  J =  1 , K234                                            HITS1542
          N2 =       N2D(I,J,J)                                           HITS1543
          NX =       NVARX(J)                                             HITS1544
2           SUM =  SUM + ( C(N2)  *B(NX))**2                              HITS1545
            SUM = SUM /2.D0                                               HITS1546
         IF(  K1 .EQ. 0  ) GO TO 4                                        HITS1547
         DO  3  J = 1, K1                                                 HITS1548
      NXJ = NVARX(J)                                                      HITS1549
         BNXJ  =  B(NXJ)                                                  HITS1550
            KB = J +1                                                     HITS1551
         DO  3  K =  KB , K234                                            HITS1552
            N2 =       N2D(I ,J ,K )                                      HITS1553
            NXK  =  NVARX(K)                                              HITS1554
3        SUM = SUM +  C(N2) ** 2  *  BNXJ   * B(NXK)                      HITS1555
4          NY =  NVAR (I)                                                 HITS1555
1          C(NY) = C(NY)  + SUM                                           HITS1557
           RETURN                                                         HITS1558
            END                                                           HITS1559
```

D.21 Subroutine G1795

```
      SUBROUTINE  G1795                                                 HITS1560
      RETURN                                                            HITS1561
      END                                                               HITS1562
```

D.22 Subroutine G2440

```
      SUBROUTINE  G2440                                        HITS1563
      RETURN                                                   HITS1564
      END                                                      HITS1565
```

D.23 Subroutine G2987

```
      SUBROUTINE G2987
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/OEC/OE(600)
      COMMON /CISPRN/ ISPRNT
      COMMON /CINOM/ INOM
      EQUIVALENCE           (XBART   ,OE( 131)), (YBART   ,OE( 132)),
     A(ZBART   ,OE( 133)), (YBARX   ,OE( 134)), (ZBARX   ,OE( 135))
      EQUIVALENCE           (IFCRW   ,OE( 201)), (IFCRC   ,OE( 202)),
     A(IFC     ,OE( 203)), (ICONV   ,OE( 205))
      EQUIVALENCE           (XBARTI  ,OE( 301)), (YBARTI  ,OE( 302)),
     A(ZBARTI  ,OE( 303)), (YBARXI  ,OE( 304)), (ZBARXI  ,OE( 305)),
     B(DXT     ,OE( 306)), (DYT     ,OE( 307)), (DZT     ,OE( 308)),
     C(DYX     ,OE( 309)), (DZX     ,OE( 310)), (DVXT    ,OE( 311)),
     D(DVYT    ,OE( 312)), (DVZT    ,OE( 313)), (DVXX    ,OE( 314)),
     E(DVYX    ,OE( 315)), (DVZX    ,OE( 316)), (DT      ,OE( 317)),
     F(RADT    ,OE( 318)), (RADX    ,OE( 319)), (RHOT    ,OE( 320)),
     G(RHOX    ,OE( 321)), (DXTI    ,OE( 322)), (DYTI    ,OE( 323)),
     H(DZTI    ,OE( 324)), (DXTE    ,OE( 325)), (DYTE    ,OE( 326)),
     I(DZTE    ,OE( 327)), (DYXI    ,OE( 328)), (DZXI    ,OE( 329)),
     J(DYXE    ,OE( 330)), (DZXE    ,OE( 331)), (DXDYT   ,OE( 332)),
     K(DXDZT   ,OE( 333)), (DYDZT   ,OE( 334)), (DYDZX   ,OE( 335)),
     L(DXDYTI  ,OE( 336)), (DXDZTI  ,OE( 337)), (DYDZTI  ,OE( 338)),
     M(DYDZXI  ,OE( 339)), (DXDYTE  ,OE( 340)), (DXDZTE  ,OE( 341)),
     N(DYDZTE  ,OE( 342)), (DYDZXE  ,OE( 343))
      EQUIVALENCE           (TOL     ,OE( 453))
      EQUIVALENCE           (TN      ,OE( 500)), (XTFC    ,OE( 501)),
     A(YTFC    ,OE( 502)), (ZTFC    ,OE( 503)), (VXTFC   ,OE( 504)),
     B(VYTFC   ,OE( 505)), (VZTFC   ,OE( 506))
      EQUIVALENCE           (XT      ,OE( 507)), (YT      ,OE( 508)),
     A(ZT      ,OE( 509)), (VXT     ,OE( 510)), (VYT     ,OE( 511)),
     B(VZT     ,OE( 512)), (YR      ,OE( 513)), (ZR      ,OE( 514)),
     C(VXR     ,OE( 515)), (VYR     ,OE( 516)), (VZR     ,OE( 517)),
     D(TX      ,OE( 518))
C
C
C**************************************************************************
C
C     FIRE CONTROL LOGIC
C
C
      IF((.NOT.((IFCRW.EQ.1).AND.(IFC.EQ.1))).AND.(IFCRC.NE.1)) GO TO 1
      DO 2 I = 1,100
    2 OE(I) = OE(I+100)
    1 CONTINUE
      IF((IFC.NE.1).AND.(IFCRC.NE.1)) GO TO 3
      TOL = 10.**ICONV
      CALL FC2987
      IF(INOM.EQ.1) GO TO 500
      IF(ISPRNT.EQ.0) GO TO 600
  500 CALL SPRNT
```

D.23 Subroutine G2987 (continued)

```
  600 CONTINUE                                                          HITS1616
  3   IFC = 0                                                           HITS1617
      CALL RW2987                                                       HITS1618
C                                                                       HITS1619
C***********************************************************************HITS1620
C                                                                       HITS1621
C     INCREMENTAL OUTPUTS                                               HITS1622
C                                                                       HITS1623
C                                                                       HITS1624
      DXT    = XT     - XTFC                                            HITS1625
      DYT    = YT     - YTFC                                            HITS1626
      DZT    = ZT     - ZTFC                                            HITS1627
      DYX    = YR     - YTFC                                            HITS1628
      DZX    = ZR     - ZTFC                                            HITS1629
      DVXT   = VXT    - VXTFC                                           HITS1630
      DVYT   = VYT    - VYTFC                                           HITS1631
      DVZT   = VZT    - VZTFC                                           HITS1632
      DVXX   = VXR    - VXTFC                                           HITS1633
      DVYX   = VYR    - VYTFC                                           HITS1634
      DVZX   = VZR    - VZTFC                                           HITS1635
      DT     = TX     - TN                                              HITS1636
      DXTI   = DXT    - XBARTI                                          HITS1637
      DYTI   = DYT    - YBARTI                                          HITS1638
      DZTI   = DZT    - ZBARTI                                          HITS1639
      DXTE   = DXT    - XBART                                           HITS1640
      DYTE   = DYT    - YBART                                           HITS1541
      DZTE   = DZT    - ZBART                                           HITS1642
      DYXI   = DYX    - YBARXI                                          HITS1643
      DZXI   = DZX    - ZBARXI                                          HITS1644
      DYXE   = DYX    - YBARX                                           HITS1645
      DZXE   = DZX    - ZBARX                                           HITS1646
C                                                                       HITS1647
C***********************************************************************HITS1648
C                                                                       HITS1649
C                                                                       HITS1650
C     SEP AND CEP CALCULATIONS USING INTERNAL AND EXTERNAL CENTERS      HITS1651
C                                                                       HITS1652
C                                                                       HITS1653
      RADT = DSQRT(DXTE**2 + DYTE**2 + DZTE**2)                         HITS1654
      RADX = DSQRT(DYXE**2 + DZXE**2)                                   HITS1655
      RHOT = DSQRT(DXTI**2 + DYTI**2 + DZTI**2)                         HITS1656
      RHOX = DSQRT(DYXI**2 + DZXI**2)                                   HITS1657
C                                                                       HITS1658
C***********************************************************************HITS1659
C                                                                       HITS1660
C     CORRELATIONS                                                      HITS1661
C                                                                       HITS1662
C                                                                       HITS1663
      DXDYT  = DXT    * DYT                                             HITS1664
      DXDZT  = DXT    * DZT                                             HITS1665
```

D.23 Subroutine G2987 (concluded)

```
      DYDZT  = DYT    * DZT                                             HITS1666
      DYDZX  = DYX    * DZX                                             HITS1667
      DXDYTI = DXTI   * DYTI                                            HITS1668
      DXDZTI = DXTI   * DZTI                                            HITS1669
      DYDZTI = DYTI   * DZTI                                            HITS1670
      DYDZXI = DYXI   * DZXI                                            HITS1671
      DXDYTE = DXTE   * DYTE                                            HITS1672
      DXDZTE = DXTE   * DZTE                                            HITS1673
      DYDZTE = DYTE   * DZTE                                            HITS1674
      DYDZXE = DYXE   * DZXE                                            HITS1675
C INOM = 1 IN MAIN.                                                     HITS1676
      IF ( INOM .EQ. 1 )  GO TO 260                                     HITS1677
      IF ( ISPRNT .EQ. 0 )  RETURN                                      HITS1678
  260 CALL SPRNT                                                        HITS1679
      INOM = 0                                                          HITS1680
      RETURN                                                            HITS1681
      END                                                               HITS1582
```

D.24 Subroutine HSTG1

```
      SUBROUTINE HSTG1(IER)                                             HITS1683
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1684
      COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS1685
     1IY,KARS1(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS1686
      COMMON/OEC/OE(600)                                                HITS1687
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS1688
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS1689
      EQUIVALENCE (KADD,KC)                                             HITS1690
C      THIS SUBROUTINE CHECKS THE SAMPLE OUT PUT AGAINST THE INTERNALLY HITS1691
C      GENERATED HISTOGRAM RANGES-IF A REJECT OCCURS IT SETS IER=-1     HITS1692
C      THIS SUBROUTINE REQUIRES KADD THE NOMINAL BLOCK ADDRESS IN C = KCHITS1693
      ENN=NCELL                                                         HITS1694
      IBN=1                                                             HITS1695
      LADD=2*KD*   K234 + KD + (KADD - 1)                               HITS1696
      DO 1 I=1,KD                                                       HITS1697
      IADD=ID(I,1)                                                      HITS1698
C      IADD IS ADDRESS OF OUTPUT VARIABLE VALUE-CALCULATE ADDRESSES     HITS1699
C      OF VARIABLE MEAN AND TOLERANCE VALUES IN C ARRAY                 HITS1700
      IMN=LADD+I                                                        HITS1701
      IVAR=IMN+KD                                                       HITS1702
      TOL=3.*DSQRT(C(IVAR))                                             HITS1703
      FJ=(OE(IADD)-C(IMN)+TOL)*.5*ENN/TOL                               HITS1704
      IF ( FJ ) 4,77,77                                                 HITS1705
   77 J=FJ                                                              HITS1706
      J=J+1                                                             HITS1707
C      J IS HISTOGRAM INTERVAL TO BE INCREMENTED-CHECK IF VALID         HITS1708
      IF(J)4,4,2                                                        HITS1709
    2  IF(J-NCELL)3,3,4                                                 HITS1710
C     IF J LT ZERO OR GT NCELL REJECT SAMPLE                            HITS1711
    4  IR1(I)=IR1(I)+1                                                  HITS1712
      IDT(I)=0                                                          HITS1713
      IER=-1                                                            HITS1714
      RETURN                                                            HITS1715
C      STORE HISTOGRAM CELL ADDRESS TEMPORARILY                         HITS1716
    3 IDT(I)=IBN+J-1                                                    HITS1717
    1  IBN=IBN+NCELL                                                    HITS1718
C      IF THIS POINT REACHED THE SAMPLE IS OK SO INCREMENT HISTOGRAMS   HITS1719
      DO 5 I=1,KD                                                       HITS1720
      L=IDT(I)                                                          HITS1721
    5  ID1(L)=ID1(L)+1                                                  HITS1722
      DO 6 I=1,K234                                                     HITS1723
      L=IIT(I)                                                          HITS1724
    6 II1(L)=II1(L)+1                                                   HITS1725
      RETURN                                                            HITS1726
      END                                                               HITS1727
```

D.25 Subroutine HSTG2

```
      SUBROUTINE HSTG2(IER)                                             HITS1728
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1729
      COMMON/CAPDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS1730
     1IY,KARS1(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS1731
      COMMON/OEC/OE(600)                                                HITS1732
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS1733
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS1734
      EQUIVALENCE (KADD,KC)                                             HITS1735
C       THIS ROUTINE CHECKS THE SAMPLE OUTPUT AGAINST THE USER SPEC     HITS1736
      ENN=NCELL                                                         HITS1737
      IBN=1                                                             HITS1738
      DO 1 I=1,KD                                                       HITS1739
      L=ID(I,2)                                                         HITS1740
C       CHECK IF IT IS A TYPE 8 VARIABLE                                HITS1741
      IF(L)8,8,2                                                        HITS1742
C      IF IT IS NOT A TYPE 8 CHECK IF IT PASSED THE PREVIOUS TIME       HITS1743
    8   IF (IDT(I))4,4,1                                                HITS1744
C       IF NOT A TYPE 8 AND REJECTED BEFORE REJECT IT AGAIN IF PASSED   HITS1745
C       BEFORE PASS IT AGAIN                                            HITS1746
    2   N=ID(I,1)                                                       HITS1747
      FJ=(OE(N)-B(L+1)+B(L+2))*.5*ENN/B(L+2)                            HITS1748
      IF ( FJ ) 4,77,77                                                 HITS1749
   77 J=FJ                                                              HITS1750
      J=J+1                                                             HITS1751
C       J IS THE HISTOGRAM INTERVAL NUMBER CHECK IF VALID               HITS1752
      IF(J)4,4,3                                                        HITS1753
    3 IF(J-NCELL)5,5,4                                                  HITS1754
    4   IR2(I)=IR2(I)+1                                                 HITS1755
      IER=-1                                                            HITS1756
      RETURN                                                            HITS1757
    5   IDT(I)=IBN+J-1                                                  HITS1758
    1   IBN=IBN+NCELL                                                   HITS1759
C       IF GETS HERE IT IS SUCCESSFUL SO INCREMENT HISTOGRAMS           HITS1760
      DO 6 I=1,KD                                                       HITS1761
      L=IDT(I)                                                          HITS1762
    6   ID2(L)=ID2(L)+1                                                 HITS1763
      DO 7 I=1,K234                                                     HITS1764
      L=IIT(I)                                                          HITS1765
    7   II2(L)=II2(L)+1                                                 HITS1765
      RETURN                                                            HITS1767
      END                                                               HITS1768
```

D.26 Subroutine *HYSPLT*

```
      SUBROUTINE HYSPLT                                                 HITS1769
       IMPLICIT  REAL * 8(A-H , O-Y)                                    HITS1770
      COMMON/ HISPLT/ IOPLOT   ,ISPLOT                                  HITS1771
     1                         , ZTITLE(20,30) , ZDATAD(21,4,30)        HITS1772
      COMMON /CARDCM / B(200) , C(5000) , IA(200,2) , ID(10,2) , INC    HITS1773
     1 ,IOPRNT , IOUT , IY ,KARSI(3) , KA , KB , KC , KD , KTI , K234   HITS1774
     2 , LABC(3)                                                        HITS1775
       NCELL1 =ZTITLE(18,1) +1                                          HITS1776
      WRITE(IOUT,1500)                                                  HITS1777
 1500 FORMAT(1H1)                                                       HITS1778
          WRITE(IOUT,1000)                                              HITS1779
 1000  FORMAT(    '    J                     ZTITLE '   )              HITS1780
       K = KD  + K234                                                   HITS1781
       DO 1          J   =  1, K                                        HITS1782
    1 WRITE(IOUT ,1001) J,(ZTITLE(I,J), I=1,20 )                        HITS1783
 1001  FORMAT('C',I5, 6X ,  1P10E12.4,/,12X,1P10E12.4)                  HITS1784
         DO    2    J =  1, K                                           HITS1785
      WRITE(IOUT ,500)                                                  HITS1786
    2  WRITE(IOUT,1002 )J ,((ZDATAD( I, L , J) ,  L = 1,4 ) , I=1,NCELL HITS1787
     11  )                                                              HITS1788
 1002  FORMAT(   ' ZDATAD     J=',I3,    '      I=1          I=2       IHITS1789
     1=3          I=4    '/  ( 15X , 1P4E12.4 )  )                      HITS1790
          RETURN                                                        HITS1791
           END                                                          HITS1792
```

D.27 Subroutine INCARD

```
      SUBROUTINE INCARD                                                 HITS1793
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS1794
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,       HITS1795
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS1796
      COMMON/OEC/OE(600)                                                HITS1797
      DIMENSION IOE(2,600)                                              HITS1798
      DIMENSION IB(2,200)                                               HITS1799
      EQUIVALENCE  (IB(1,1), B(1)), (IOE(1,1), OE(1))                   HITS1800
      DO 15 I=1,3                                                       HITS1801
   15 LABC(I) = 1                                                       HITS1802
C KT1 WILL COUNT TYPE 1.  K234 WILL COUNT TYPES 2, 3 AND 4.             HITS1803
C KD FOR THE DEPENDENT VARIABLES.                                       HITS1804
C KB FOR THE CURRENT ENTRY IN B.                                        HITS1805
C KC FOR THE CURRENT ENTRY IN C.                                        HITS1806
      KT1 = 0                                                           HITS1807
      K234 = 0                                                          HITS1808
      KD = 0                                                            HITS1809
      KB = 0                                                            HITS1810
      KC = 0                                                            HITS1811
      IEXIT = 0                                                         HITS1812
   20 READ (INC,9002)  ISOE, ITYPE, FNOM, TOL, STDEV, ITL               HITS1813
 9002 FORMAT(I5, I2, 3F13.1, I4)                                        HITS1814
      IF ( ISOE .LT. 0 )  GO TO 1100                                    HITS1815
      WRITE(IOUT,5000) ISOE, ITYPE, FNOM, TOL, STDEV, ITL               HITS1816
 5000 FORMAT(10X,5X,I5,6X,7X,I2,7X,3(3X,1PE11.4,2X),6X,I4,/)            HITS1817
      VAR = STDEV * STDEV                                               HITS1818
      GO TO (100, 203, 203, 400, 500, 999, 708, 708), ITYPE             HITS1819
  100 KT1 = KT1 + 1                                                     HITS1820
      IF ( KT1 .LE. 3 )  GO TO 110                                      HITS1821
      WRITE (IOUT,9004)  ISOE                                           HITS1822
 9004 FORMAT(26H MORE THAN 3 TYPE 1, NAME=  , I4)                       HITS1823
      IEXIT = 1                                                         HITS1824
      GO TO 1080                                                        HITS1825
  110 ITLADD = 1                                                        HITS1826
      GO TO 1010                                                        HITS1827
C 120 SAVE THE KAR SUBSCRIPT FOR LATER MOVING THE TYPE 1 INFO UP TO     HITS1828
C   THE START OF IA.                                                    HITS1829
  120 KARS1(KT1) = KAR                                                  HITS1830
      LABC(KT1) = ITL                                                   HITS1831
      GO TO 20                                                          HITS1832
  203 K234 = K234 + 1                                                   HITS1833
      GO TO 110                                                         HITS1834
C 210 SAVE 2 SLOTS IN IB (FOR TABLE LENGTH AND ADDRESS IN C).           HITS1835
  210 KB = KB + 2                                                       HITS1836
      GO TO 710                                                         HITS1837
  400 K234 = K234 + 1                                                   HITS1838
C SKIP 3 SLOTS FOR NOM, TOL AND VAR.                                    HITS1839
      ITLADD = 4                                                        HITS1840
C I4 KEEPS TRACK OF READING THE W AND Y TABLES.                         HITS1841
      I4 = 1                                                            HITS1842
```

D.27 Subroutine INCARD (continued)

```
      GO TO 1010                                                     HITS1843
  410 IF ( I4 .EQ. 2 )  GO TO 420                                    HITS1844
      I4 = 2                                                         HITS1845
      KC = KC + 1                                                    HITS1846
      GO TO 1070                                                     HITS1847
  420 IAW = IB(1,KB) - 1                                             HITS1848
      IAY = IAW + ITL                                                HITS1849
      KBR = KB - 5                                                   HITS1850
      CALL T4NTV ( IAW, IAY, ITL, KBR )                              HITS1851
C PUT NOMINAL VALUE INTO OE.                                         HITS1852
      LOE = IA(KAR,1)                                                HITS1853
      OE(LOE) = B(KBR+1)                                             HITS1854
      GO TO 20                                                       HITS1855
  500 GO TO 1010                                                     HITS1856
C 510 PUT A -1 IN COLUMN 2 FOR A LATER CHECK.                        HITS1857
  510 IA(KAR,2) = -1                                                 HITS1858
C IS IT AN INTEGER.                                                  HITS1859
      LOE = IA(KAR,1)                                                HITS1860
      IF ( LOE .GT. 0 )  GO TO 710                                   HITS1861
      LOE = - LOE                                                    HITS1862
      IOE(1,LOE) = FNOM                                              HITS1863
      GO TO 20                                                       HITS1864
  708 KD = KD + 1                                                    HITS1865
      ID(KD,1) = ISOE                                                HITS1866
      IF ( ITYPE .EQ. 7 )  GO TO 20                                  HITS1867
      KB = KB + 1                                                    HITS1868
      ID(KD,2) = KB                                                  HITS1869
      ITLADD = 1                                                     HITS1870
      GO TO 1045                                                     HITS1871
  710 OE(ISOE) = FNOM                                                HITS1872
      GO TO 20                                                       HITS1873
  999 WRITE (IOUT,1000)                                              HITS1874
 1000 FORMAT(25H INCARD ERROR, CALL EXIT.)                           HITS1875
      CALL EXIT                                                      HITS1876
 1010 KAR = 0                                                        HITS1877
 1020 KAR = KAR + 1                                                  HITS1878
      IF ( IABS(IA(KAR,1) ) .EQ. ISOE )  GO TO 1030                  HITS1879
      IF ( KAR .LT. KA )  GO TO 1020                                 HITS1880
      IEXIT = 1                                                      HITS1881
      WRITE (IOUT,9003)  ISOE, ITYPE                                 HITS1882
 9003 FORMAT(24H INCARD, COULD NOT FIND , I4, 16H IN IA FOR TYPE , I2)  HITS1883
      IF ( ITYPE .EQ. 1  .OR.  ITYPE .EQ. 4 )  GO TO 1080            HITS1884
      GO TO 20                                                       HITS1885
 1030 GO TO (1040, 1040, 1040, 1040, 510, 999, 999, 999), ITYPE      HITS1886
 1040 KB = KB + 1                                                    HITS1887
      IA(KAR,2) = KB                                                 HITS1888
 1045 IB(1,KB) = ITYPE                                               HITS1889
      KB = KB + ITLADD                                               HITS1890
      GO TO ( 1060, 1050, 1050, 1060, 999, 999, 999, 1050), ITYPE    HITS1891
 1050 B(KB) = FNOM                                                   HITS1892
```

D.27 Subroutine INCARD (concluded)

```
      KB = KB + 1                                                       HITS1893
      B(KB) = TOL                                                       HITS1894
      KB = KB + 1                                                       HITS1895
      B(KB) = VAR                                                       HITS1896
      GO TO ( 999, 210, 210, 999, 999, 999, 999, 710), ITYPE           HITS1897
 1060 IB(1,KB) = ITL                                                    HITS1898
      KC = KC + 1                                                       HITS1899
      KB = KB + 1                                                       HITS1900
      IB(1,KB) = KC                                                     HITS1901
 1070 KCM1 = KC - 1                                                     HITS1902
      READ (INC,9005)  (C(I+KCM1), I=1,ITL)                             HITS1903
 9005 FORMAT(5F13.1)                                                    HITS1904
C THE LAST ADDRESS USED IS KCM1 + ITL.                                  HITS1905
      KC = KCM1 + ITL                                                   HITS1906
      GO TO ( 120, 999, 999, 410, 999, 999, 999, 999), ITYPE           HITS1907
 1080 READ (INC,9005)  (C(I), I=1,ITL)                                  HITS1908
      IF ( ITYPE .EQ. 1 )  GO TO 20                                     HITS1909
C FOR ITYPE=4 WE HAVE TO BYPASS BOTH W AND Y TABLES.                    HITS1910
      ITYPE = 1                                                         HITS1911
      GO TO 1080                                                        HITS1912
 1100 CONTINUE                                                          HITS1913
C FINISHED READING ALL THE INPUT CARDS. WERE THERE ANY ERRORS.          HITS1914
      IF ( IEXIT .EQ. 1 )  CALL EXIT                                    HITS1915
      RETURN                                                            HITS1916
      END                                                               HITS1917
```

D.28 Subroutine INCON

```
      SUBROUTINE  INCON                                                 HITS1918
        IMPLICIT REAL * 8 (A-H, O -Z )                                  HITS1919
 COMMON/DEC/DE(600)                                                     HITS1920
 EQUIVALENCE               (VO      ,DE( 101)), (VWIND  ,DE( 102)),     HITS1921
A(PHIWIN ,DE( 103)), (RHO     ,DE( 104)), (CD1    ,DE( 105)),           HITS1922
B(V1     ,DE( 106)), (CD2     ,DE( 107)), (V2     ,DE( 108)),           HITS1923
C(XN     ,DE( 109)), (CNA     ,DE( 110)), (SM     ,DE( 111)),           HITS1924
D(CMQ    ,DE( 112)), (CMPA    ,DE( 113)), (TRIM   ,DE( 114)),           HITS1925
E(PNITRI ,DE( 115)), (ANGO    ,DE( 116)), (PHIANG ,DE( 117)),           HITS1926
F(RATEO  ,DE( 118)), (PHIRAT  ,DE( 119)), (GAMO   ,DE( 120)),           HITS1927
G(PHIGAM ,DE( 121)), (A       ,DE( 122)), (ELL    ,DE( 123)),           HITS1928
H(D      ,DE( 124)), (W       ,DE( 125)), (AIX    ,DE( 126)),           HITS1929
I(AIY    ,DE( 127)), (P       ,DE( 128)), (ALPCON ,DE( 129)),           HITS1930
J(CAA    ,DE( 130))                                                     HITS1931
  EQUIVALENCE              (IDUMP   ,DE( 204)), (ICNCL  ,DE( 206))      HITS1932
  EQUIVALENCE              (ALPHA   ,DE( 400)), (ATOT   ,DE( 401)),     HITS1933
A(ALPTRM ,DE( 402)), (ALPHDO  ,DE( 403)), (B      ,DE( 404)),           HITS1934
B(BETTRM ,DE( 405)), (BETAO   ,DE( 406)), (BETADO ,DE( 407)),           HITS1935
C(BETA   ,DE( 408)), (ALPMAX  ,DE( 409)), (ALPMIN ,DE( 410)),           HITS1936
D(CAYD   ,DE( 411)), (CD8     ,DE( 412)), (CMA    ,DE( 413)),           HITS1937
E(CMTHTD ,DE( 414)), (CMTHTA  ,DE( 415)), (CAY1   ,DE( 416)),           HITS1938
F(CAY2   ,DE( 417)), (CDAOB   ,DE( 418)), (CDAB   ,DE( 419)),           HITS1939
G(DELTT  ,DE( 420)), (DELV    ,DE( 421)), (DELT   ,DE( 422)),           HITS1940
H(DELX   ,DE( 423)), (DYDXO   ,DE( 424)), (DZDXO  ,DE( 425)),           HITS1941
I(DYDTO  ,DE( 426)), (DZDTO   ,DE( 427)), (DELW   ,DE( 428)),           HITS1942
J(DELLAM ,DE( 429)), (EYEP    ,DE( 430)), (EMP    ,DE( 431)),           HITS1943
K(EOLT   ,DE( 432)), (EDLT    ,DE( 433)), (F      ,DE( 434)),           HITS1944
L(M      ,DE( 435)), (PSIDO   ,DE( 436)), (PHIO   ,DE( 437)),           HITS1945
M(PHI1   ,DE( 438)), (PHI2    ,DE( 439)), (PSIO   ,DE( 440)),           HITS1946
N(R1     ,DE( 441)), (R2      ,DE( 442)), (R3     ,DE( 443)),           HITS1947
O(R4     ,DE( 444)), (RTRIM   ,DE( 445)), (TC     ,DE( 446)),           HITS1948
P(TCO    ,DE( 447)), (TS      ,DE( 448)), (TGO    ,DE( 449)),           HITS1949
Q(TP     ,DE( 450)), (THETAO  ,DE( 451)), (THETDO ,DE( 452)),           HITS1950
R(TOL    ,DE( 453)), (VC      ,DE( 454)), (VCO    ,DE( 455)),           HITS1951
S(VYA    ,DE( 456)), (VZA     ,DE( 457)), (WX     ,DE( 458))            HITS1952
  EQUIVALENCE              (WZ      ,DE( 459)), (WO     ,DE( 460)),     HITS1953
A(W1     ,DE( 461)), (W2      ,DE( 462)), (WL2    ,DE( 463)),           HITS1954
B(XLAMO  ,DE( 464)), (XLAM1   ,DE( 465)), (XLAM2  ,DE( 466)),           HITS1955
C(XMU1   ,DE( 467)), (XMU2    ,DE( 468)), (XG     ,DE( 469)),           HITS1956
D(XMU    ,DE( 470)), (XJAY    ,DE( 471)), (XJAZ   ,DE( 472)),           HITS1957
E(XJA    ,DE( 473)), (YA      ,DE( 474)), (ZA     ,DE( 475)),           HITS1958
F(ALPHO  ,DE( 476))                                                     HITS1959
    DATA  RFAC  , G                   / 57.2957795100 , 32.174D0 /      HITS1960
 WX = VWIND                                                             HITS1961
 WZ = PHIWIN                                                            HITS1962
      V12 = V1 * V1                                                     HITS1963
   CAYD =  (CD2 - CD1 )  /( 1.D0/       V2/V2- 1.D0 / V12)              HITS1964
   CD8  =  CD1   -CAYD / V12                                            HITS1965
     B  =  1.D0   + ( WZ / (VO - WX ) ) **  2                           HITS1966
     F  =   RHO *  A  *  G       /2.D0 / W                              HITS1967
```

D.28 Subroutine INCON (continued)

```
      IF(IDUMP    .EQ.1)WRITE(6,1000)WX,WZ,CAYD, CD8, B, F              HITS1968
 1000 FORMAT(// '       WX              WZ             CAYD           CD8 8HITS1969
     1          F      ' /1X,1P6E12.4 /)                                 HITS1970
      THETA0 = ANGO/RFAC                                                 HITS1971
      PSI0 = PHIANG/RFAC                                                 HITS1972
      THETD0 = RATE0                                                     HITS1973
      PSID0 = PHIRAT                                                     HITS1974
      DYDX0 = GAM0/RFAC                                                  HITS1975
      DZDX0 = PHIGAM/RFAC                                                HITS1976
          DYDT0    =      V0   *    DYDX0                                HITS1977
          DZDT0    =      V0   *    DZDX0                                HITS1978
        ALPH0     =   THETA0   - DYDX0                                   HITS1979
        BETA0     =    DZDX0   -   PSI0                                  HITS1980
      IF(IDUMP    .EQ.1)WRITE(6,1001)                 THETA0,PSI0,THETD0, HITS1981
     1 PSID0, DYDX0 , DZDX0 , DYDT0 , DZDT0                              HITS1982
           FCV   =   F *   CNA * V0                                      HITS1983
         ALPHD0 = THETD0 - FCV   * ALPH0 - P * BETA0                     HITS1984
      IF(ICNCL.EQ.1)ALPHD0=ALPHD0+F*V0*THETA0*(CD8+CAYD/V0/V0)           HITS1985
        BETAD0  =   -PSID0- FCV   * BETA0 + P * ALPH0                    HITS1986
      IF(ICNCL.EQ.1)BETAD0=BETAD0-F*V0*PSI0*(CD8+CAYD/V0/V0)             HITS1987
        CMA   =   - CNA * ELL  *  SM /   D                               HITS1988
         EYEP=   2.D0 *   AIY /RHO / A /D /V0/V0                         HITS1989
        EMP   =   2.D0 * W / RHO /A   / G   / V0                         HITS1990
         D2V  =   D /2.D0 /V0                                            HITS1991
        CMTHTD =   CMQ * D2V                                             HITS1992
        CMTHTA =   CMPA* D2V                                             HITS1993
           H   =  ZZ(CNA, CD8, CAYD, V0, EMP, ICNCL)                     HITS1994
         CCON =       CNA / EMP   + CMTHTD / EYEP                        HITS1995
           PII=   P   * AIX / AIY                                        HITS1996
        WCON = -4.D0 * CMA / EYEP   + PII * PII   -CCON * CCON           HITS1997
      IF(IDUMP    .EQ. 1)WRITE(6,1002)ALPH0,BETA0,ALPHD0,BETAD0, CMA,EYEPHITS1998
     1, EMP, CMTHTD, CMTHTA, H                                           HITS1999
           W0   = DSQRT(.125   * ( WCON   + DSQRT( WCON * WCON + 4.D0*(PIIHITS2000
     1 * CCON + 2.D0 * P     * CMTHTA / EYEP ) ** 2 )     )   )          HITS2001
         IF(   CMA .LT.   0.) GO TO 3                                    HITS2002
          PCR=0.D0                                                       HITS2003
          XMU=0.D0                                                       HITS2004
         RTRIM=1.D0                                                      HITS2005
        PHIROL= 0.D0                                                     HITS2006
       GO TO 4                                                           HITS2007
 3        CONTINUE                                                       HITS2008
         PCRV= DSQRT(-CMA*RHO *A /2.D0*D/(AIY-AIX))                      HITS2009
        PCR= PCRV*DABS(V0)                                               HITS2010
        XMU=( CNA*RHO/4.D0  *A/W*G-CMQ*D*D/8.D0*RHO*A/(AIY-AIX))/PCRV    HITS2011
          PPCR= P/PCR                                                    HITS2012
        PPCR1=1.D0- PPCR**2                                              HITS2013
         XMU2=2.D0 *XMU*PPCR                                             HITS2014
        RTRIM= 1.D0 /DSQRT(PPCR1**2 + XMU2**2)                           HITS2015
          PHIROL= DATAN(XMU2/PPCR1)                                      HITS2016
          IF( PPCR1 .LT. 0.) PHIROL= PHIROL+3.14159265358980D0           HITS2017
```

D.28 Subroutine INCON (continued)

```
      ABAR = DSQRT(TRIM**2 + PNITRI**2)                                 HITS2018
      PNITRR = ZATAN2(PNITRI,TRIM)                                      HITS2019
    4       CONTINUE                                                    HITS2020
             ALPTRM = RTRIM * ABAR /RFAC                                HITS2021
             PTPR   =  PNITRR + PHIROL                                  HITS2022
             BETTRM =  ALPTRM * DSIN( PTPR )                            HITS2023
             ALPTRM =  ALPTRM * DCOS( PTPR )                            HITS2024
          IF(IDUMP    .EQ.  1 ) WRITE(6,1006) XMU ,RTRIM,PHIROL,ALPTRM  HITS2025
     1 , BETTRM  ,PCR                                                   HITS2026
 1006  FORMAT(/  '       XMU        RTRIM       PHIROL      ALPTRM     BETHITS2027
     1TRM        PCR'/                                                  HITS2028
     2       1X,  1P10E12.4 //)                                         HITS2029
             DELW  =  P  - PII / 2.D0                                   HITS2030
           XLAMO = .5D0* (CMTHTD /EYEP -CNA/EMP )                       HITS2031
           DELLAM = PII /4.D0/ W0 *     CCON  + P * CMTHTA/2.D0/W0 /EYEP HITS2032
           W1 = W0 - DELW                                               HITS2033
           W2 = W0 + DELW                                               HITS2034
             AA =  ALPHO  -ALPTRM                                       HITS2035
           XLAM1 =  XLAMO    + DELLAM                                   HITS2036
           XLAM2 =  XLAMO    - DELLAM                                   HITS2037
             BB = BETAO  -BETTRM                                        HITS2038
           RCON1 =  ALPHDO + W2 * BB                 -XLAM2 * AA        HITS2039
           RCON2 =  BETADO- W2 * AA    -XLAM2 * BB                      HITS2040
           WL2   = 2.D0 * (W0 * W0 + DELLAM ** 2 )                      HITS2041
             R1 = ( W0 * RCON1  +  DELLAM * RCON2) / WL2                HITS2042
             R2 = (-W0 * RCON2  +  DELLAM * RCON1 ) / WL2               HITS2043
      IF(IDUMP   .EQ.1)WRITE(6,1004) W0, DELW, XLAMO , DELLAM , W1, W2  HITS2044
     1,XLAM1, XLAM2, WL2  , R1, R2                                      HITS2045
 1004 FORMAT(    '      W0          DELW        XLAMO       DELLAM      WHITS2046
     11         W2          XLAM1       XLAM2       WV2         R1      HITS2047
     2       R2   '/1X,   1P11E12.4   // )                              HITS2048
              R4 = AA - R2                                              HITS2049
              R3= BB -R1                                                HITS2050
           CAY1 =  DSQRT( R1* R1  + R2 * R2 )                           HITS2051
           CAY2 =  DSQRT( R3* R3  + R4 * R4 ;                           HITS2052
             XNU1 = ZATAN2( R2 , R1 )                                   HITS2053
             XNU2 = ZATAN2(R4 , R3 )                                    HITS2054
             PHI1 = DATAN2(-W1 - P, XLAM1 )                             HITS2055
             PHI2 = DATAN2(-W2 + P, XLAM2 )                             HITS2056
             PHIO = DATAN2( -W0 ,  XLAMO    )                           HITS2057
 1001 FORMAT(                              '   THETAO      PSIO      THEHITS2058
     1TDO        PSIDO       DYDX0       DZDX0       DYDTO       DZDTO'  HITS2059
     2                                              /1X,1P9E12.4 )HITS2060
 1002 FORMAT(//  '    ALPHO       BETAO       ALPHDO      BETADC     CHITS2061
     1MA         EYEP        EMP         CMTHTD      CMTHTA         H'  HITS2062
     2/1X, 1P11E12.4 /)                                                 HITS2063
             IF(IDUMP    .EQ.  1 ) WRITE(6,1005)  R3 , R4  , CAY1 , CAY2 HITS2064
     1 , XNU1 ,   XNU2 ,    PHI1  , PHI2 , PHIO                         HITS2065
 1005  FORMAT(    '       R3          R4          CAY1        CAY2     XNHITS2066
     1U1         XNU2        PHI1        PHI2        PHIO  '/1X,1P10E12.HITS2067
```

D.28 Subroutine INCON (concluded)

```
24    // )                                                        HITS2068
         RETURN                                                   HITS2069
           END                                                    HITS2070
```

D.29 Subroutine INCONF

```
      SUBROUTINE   INCONF                                                 HITS2071
        IMPLICIT REAL * 8 (A-H, O -Z )                                    HITS2072
      COMMON/OEC/OE(600)                                                  HITS2073
      EQUIVALENCE              (VO      ,OE(    1)), (VWIND  ,OE(   2)),  HITS2074
     A(PHIWIN ,OE(   3)), (RHO     ,OE(    4)), (CD1    ,OE(   5)),      HITS2075
     B(V1     ,OE(   6)), (CD2     ,OE(    7)), (V2     ,OE(   8)),      HITS2076
     C(XN     ,OE(   9)), (CNA     ,OE(   10)), (SM     ,OE(  11)),      HITS2077
     D(CMQ    ,OE(  12)), (CMPA    ,OE(   13)), (TRIM   ,OE(  14)),      HITS2078
     E(PNITRI ,OE(  15)), (ANGO    ,OE(   16)), (PHIANG ,OE(  17)),      HITS2079
     F(RATEO  ,OE(  18)), (PHIRAT  ,OE(   19)), (GAMO   ,OE(  20)),      HITS2080
     G(PHIGAM ,OE(  21)), (A       ,OE(   22)), (ELL    ,OE(  23)),      HITS2081
     H(D      ,OE(  24)), (W       ,OE(   25)), (AIX    ,OE(  26)),      HITS2082
     I(AIY    ,OE(  27)), (P       ,OE(   28)), (ALPCON ,OE(  29)),      HITS2083
     J(CAA    ,OE(  30))                                                  HITS2084
      EQUIVALENCE              (IDUMP   ,OE( 204)), (ICNCL  ,OE( 206))   HITS2085
      EQUIVALENCE              (ALPHA   ,OE( 400)), (ATOT   ,OE( 401)),  HITS2086
     A(ALPTRM ,OE( 402)), (ALPHDO  ,OE( 403)), (B      ,OE( 404)),      HITS2087
     B(BETTRM ,OE( 405)), (BETAO   ,OE( 406)), (BETADO ,OE( 407)),      HITS2088
     C(BETA   ,OE( 408)), (ALPMAX  ,OE( 409)), (ALPMIN ,OE( 410)),      HITS2089
     D(CAYD   ,OE( 411)), (CD8     ,OE( 412)), (CMA    ,OE( 413)),      HITS2090
     E(CMTHTD ,OE( 414)), (CMTHTA  ,OE( 415)), (CAY1   ,OE( 416)),      HITS2091
     F(CAY2   ,OE( 417)), (CDAOB   ,OE( 418)), (CDAB   ,OE( 419)),      HITS2092
     G(DELTT  ,OE( 420)), (DELV    ,OE( 421)), (DELT   ,OE( 422)),      HITS2093
     H(DELX   ,OE( 423)), (DYDXO   ,OE( 424)), (DZDXO  ,OE( 425)),      HITS2094
     I(DYDTO  ,OE( 426)), (DZDTO   ,OE( 427)), (DELW   ,OE( 428)),      HITS2095
     J(DELLAM ,OE( 429)), (EYEP    ,OE( 430)), (EM     ,OE( 431)),      HITS2096
     K(EOLT   ,OE( 432)), (EDLT    ,OE( 433)), (F      ,OE( 434)),      HITS2097
     L(H      ,OE( 435)), (PSIDO   ,OE( 436)), (PHIO   ,OE( 437)),      HITS2098
     M(PHI1   ,OE( 438)), (PHI2    ,OE( 439)), (PSIO   ,OE( 440)),      HITS2099
     N(R1     ,OE( 441)), (R2      ,OE( 442)), (R3     ,OE( 443)),      HITS2100
     O(R4     ,OE( 444)), (RTRIM   ,OE( 445)), (TC     ,OE( 446)),      HITS2101
     P(TCO    ,OE( 447)), (TS      ,OE( 448)), (TGO    ,OE( 449)),      HITS2102
     Q(TP     ,OE( 450)), (THETAO  ,OE( 451)), (THETDO ,OE( 452)),      HITS2103
     R(TOL    ,OE( 453)), (VC      ,OE( 454)), (VCO    ,OE( 455)),      HITS2104
     S(VYA    ,OE( 456)), (VZA     ,OE( 457)), (WX     ,OE( 458))       HITS2105
      EQUIVALENCE              (WZ      ,OE( 459)), (WO     ,OE( 460)),  HITS2106
     A(W1     ,OE( 461)), (W2      ,OE( 462)), (WL2    ,OE( 463)),      HITS2107
     B(XLAMO  ,OE( 464)), (XLAM1   ,OE( 465)), (XLAM2  ,OE( 466)),      HITS2108
     C(XNU1   ,OE( 467)), (XNU2    ,OE( 468)), (XG     ,OE( 469)),      HITS2109
     D(XMU    ,OE( 470)), (XJAY    ,OE( 471)), (XJAZ   ,OE( 472)),      HITS2110
     E(XJA    ,OE( 473)), (YA      ,OE( 474)), (ZA     ,OE( 475)),      HITS2111
     F(ALPHO  ,OE( 476))                                                  HITS2112
        DATA  RFAC  , G               / 57.2957795100 , 32.174D0 /        HITS2113
      WX = VWIND                                                          HITS2114
      WZ = PHIWIN                                                         HITS2115
          V12 = V1 * V1                                                   HITS2116
        CAYD =  (CD2 - CD1 )  /( 1.D0/       V2/V2- 1.D0 / V12)           HITS2117
        CD8  =  CD1   -CAYD / V12                                         HITS2118
          B  =  1.D0    + ( WZ / (VO - W   ) ) **  2                      HITS2119
          F  =    RHO *  A  *  G      /2.D0 / W                           HITS2120
```

D.29 Subroutine INCONF (continued)

```
      IF(IDUMP   .EQ.1)WRITE(6,1000)WX,WZ,CAYD, CD8, B, F            HITS2121
 1000 FORMAT(// '      WX            WZ          CAYD           CD8 BHITS2122
     1        F       ' /1X,1P6E12.4 /)                              HITS2123
      THETA0 = ANG0/RFAC                                             HITS2124
      PSI0 = PHIANG/RFAC                                             HITS2125
      THETD0 = RATE0                                                 HITS2126
      PSID0 = PHIRAT                                                 HITS2127
      DYDX0 = GAM0/RFAC                                              HITS2128
      DZDX0 = PHIGAM/RFAC                                            HITS2129
           DYDT0    =     V0   *    DYDX0                            HITS2130
           DZDT0    =     V0   *    DZDX0                            HITS2131
         ALPH0     =   THETA0   - DYDX0                              HITS2132
         BETA0     =   DZDX0   -   PSI0                              HITS2133
      IF(IDUMP   .EQ.1)WRITE(6,1001)                 THETA0,PSI0,THETD0, HITS2134
     1 PSID0, DYDX0 , DZDX0 , DYDT0 , DZDT0                          HITS2135
            FCV   =   F *   CNA * V0                                 HITS2136
          ALPHD0 = THETD0 - FCV   * ALPH0 - P * BETA0                HITS2137
      IF(ICNCL.EQ.1)ALPHD0=ALPHD0+F*V0*THETA0*(CD8+CAYD/V0/V0)       HITS2138
         BETAD0   =   -PSID0- FCV   * BETA0 + P * ALPH0              HITS2139
      IF(ICNCL.EQ.1)BETAD0=BETAD0-F*V0*PSI0*(CD8+CAYD/V0/V0)         HITS2140
         CMA   =   - CNA * ELL * SM /   D                            HITS2141
          EYEP=   2.D0 *   AIY /RHO / A /D /V0/V0                    HITS2142
         EMP  =   2.D0 * W / RHO /A   / G   / V0                     HITS2143
          D2V =   D /2.D0 /V0                                        HITS2144
         CMTHTD =   CMQ * D2V                                        HITS2145
         CMTHTA =   CMPA* D2V                                        HITS2146
           H   =   ZZ(CNA, CD8, CAYD, V0, EMP, ICNCL)                HITS2147
          CCON =       CNA / EMP   + CMTHTD / EYEP                   HITS2148
           PII=   P   * AIX / AIY                                    HITS2149
         WCON = -4.D0 * CMA / EYEP   + PII * PII   -CCON * CCON      HITS2150
      IF(IDUMP   .EQ. 1)WRITE(6,1002)ALPH0,BETA0,ALPHD0,BETAD0, CMA,EYEPHITS2151
     1, EMP, CMTHTD, CMTHTA, H                                       HITS2152
            W0   = DSQRT(.125   * ( WCON   + DSQRT( WCON * WCON + 4.D0*(PIIHITS2153
     1 * CCON + 2.D0 * P    * CMTHTA / EYEP ) ** 2 )    )   )        HITS2154
        IF(   CMA  .LT.   0.) GO TO 3                                HITS2155
         PCR=0.D0                                                    HITS2156
         XMU=0.D0                                                    HITS2157
        RTRIM=1.[                                                    HITS2158
       PHIROL= 0.                                                    HITS2159
      GO TO 4                                                        HITS2160
    3    CONTINUE                                                    HITS2161
        PCRV= DSQRT(-CMA*RHO *A /2.D0*D/(AIY-AIX))                   HITS2162
       PCR= PCRV*DABS(V0)                                            HITS2163
       XMU=( CNA*RHO/4.D0  *A/W*G-CMQ*D*D/8.D0*RHO*A/(AIY-AIX))/PCRV HITS2164
         PPCR= P/PCR                                                 HITS2165
       PPCR1=1.D0- PPCR**2                                           HITS2166
        XMU2=2.D0 *XMU*PPCR                                          HITS2167
       RTRIM= 1.D0 /DSQRT(PPCR1**2 + XMU2**2)                        HITS2168
         PHIROL= DATAN(XMU2/PPCR1)                                   HITS2169
         IF( PPCR1  .LT.  0.) PHIROL= PHIROL+3.14159265358980D0      HITS2170
```

D.29 Subroutine INCONF (continued)

```
      ABAR = DSQRT(TRIM**2 + PNITRI**2)                                 HITS2171
      PNITRR = ZATAN2(PNITRI,TRIM)                                      HITS2172
4         CONTINUE                                                      HITS2173
           ALPTRM = RTRIM * ABAR /RFAC                                  HITS2174
           PTPR   =  PNITPR + PHIROL                                    HITS2175
           BETTRM = ALPTRM * DSIN( PTPR )                               HITS2176
           ALPTRM = ALPTRM * DCOS( PTPR )                               H TS2177
        IF(IDUMP    .EQ.  1 ) WRITE(6,1006) XMU ,RTRIM,PHIROL,ALPTRM    HITS2178
     1 , BETTRM   ,PCR                                                  HITS2179
 1006  FORMAT(/   '         XMU          RTRIM        PHIROL       ALPTRM       BETHITS2180
     1TRM          PCR'/                                                HITS2181
     2          1X,  1P10E12.4 //)                                      HITS2182
           DELW   =  P   - PII / 2.D0                                   HITS2183
          XLAM0 = .5D0* (CMTHTD /EYEP -CNA/EMP )                        HITS2184
          DELLAM = PII /4.D0/ W0 *      CCON  + P * CMTHTA/2.D0/W0 /EYEP HITS2185
          W1 = W0 - DELW                                                HITS2186
          W2 = W0 + DELW                                                HITS2187
           AA =  ALPHO  -ALPTRM                                         HITS2188
          XLAM1 =  XLAM0   + DELLAM                                     HITS2189
          XLAM2 =  XLAM0   - DELLAM                                     HITS2190
           BB = BETA0  -BETTRM                                          HITS2191
          RCON1 =  ALPHD0 + W2 * BB              -XLAM2 * AA            HITS2192
          RCON2 =  BETAD0- W2 * AA   -XLAM2 * BB                        HITS2193
          WL2  = 2.D0 * (W0 * W0 + DELLAM ** 2 )                        HITS2194
           R1 = ( W0 * RCON1  +  DELLAM * RCON2) / WL2                  HITS2195
           R2 = (-W0 * RCON2  +  DELLAM * RCON1) / WL2                  HITS2196
       IF(IDUMP    .EQ.1)WRITE(6,1004) W0, DELW, XLAM0 , DELLAM , W1, W2 HITS2197
     1,XLAM1, XLAM2, WL2  , R1, R2                                      HITS2198
 1004 FORMAT(      '       W0          DELW         XLAM0        DELLAM       WHITS2199
     11        W2          XLAM1        XLAM2        WV2          R1        HITS2200
     2        R2  '/1X,   1P11E12.4   // )                              HITS2201
            R4 = AA - R2                                                HITS2202
            R3= BB -R1                                                  HITS2203
          CAY1 =  DSQRT( R1* R1  + R2 * R2 )                            HITS2204
          CAY2 =  DSQRT( R3* R3  + R4 * R4 )                            HITS2205
           XNU1 = ZATAN2( R2 , R1 )                                     HITS2206
           XNU2 = ZATAN2(R4 , R3 )                                      HITS2207
           PHI1 = DATAN2(-W1 - P, XLAM1 )                               HITS2208
           PHI2 = DATAN2(-W2 + P, XLAM2 )                               HITS2209
           PHI0 = DATAN2( -W0 ,  XLAM0  )                               HITS2210
 1001 FORMAT(                                  '    THETA0        PSI0       THEHITS2211
     1TD0          PSID0         DYDX0         DZDX0        DYDT0        DZDT0'  HITS2212
     2                                                  /1X,1P9E12.4 )HITS2213
 1002 FORMAT(//  '      ALPHO         BETA0         ALPHD0        BETAD0        CHITS2214
     1MA          EYEP          EMP          CMTHTD        CMTHTA         H'   HITS2215
     2/1X, 1P11E12.4 /)                                                 HITS2216
            IF(IDUMP    .EQ.  1 ) WRITE(6,1005)  R3 , R4   , CAY1 , CAY2 HITS2217
     1 , XNU1 ,   XNU2 ,     PHI1   , PHI2 , PHI0                       HITS2218
 1005  FORMAT(   '      R3            R4           CAY1          CAY2        XNHITS2219
     1U1          XNU2          PHI1          PHI2          PHI0  '/1X,1P10E12.HITS2220
```

D.29 Subroutine INCONF (concluded)

```
   24    // )                                                      HITS2221
          RETURN                                                   HITS2222
          END                                                      HITS2223
```

D.30 Subroutine INITIL

```
      SUBRO' .NE INITIL                                                 HITS2224
      IMPLI..T REAL*8 (A-H,O-Y)                                         HITS2225
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,      HITS2226
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS2227
      COMMON /CISPRN / ISPRNT                                           HITS2228
      COMMON /CMCPRN/ MCPRNT                                            HITS2229
      COMMON /CRAN/  IRANNO                                             HITS2230
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS2231
     1IRI(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS2232
      COMMON/HISPLT/ IOPLOT,ISPLOT,ZTITLE(20,30),ZDATAD(21,4,30)        HITS2233
      DIMENSION  IB(2,200), IOEA(160), IOEB(150), IOEC(150), IOE(460),  HITS2234
     1 KOE(3)                                                           HITS2235
      EQUIVALENCE  (IOE(1), IOEA(1)), (IOE(161), IOEB(1)),              HITS2236
     1 (IOE(311), IOEC(1)), (IB(1,1), B(1))                             HITS2237
      DATA IOEA     /     1,      2,      3,      4,      5,      6,      7, HITS2238
     A     8,      9,     10,     11,     12,     13,     14,     15,     16, HITS2239
     B    17,     18,     19,     20,     21,     22,     23,     24,     25, HITS2240
     C    26,     27,     28,     29,     30,    101,    102,    103,    104, HITS2241
     D   105,    106,    107,    108,    109,    110,    111,    112,    113, HITS2242
     E   114,    115,    116,    117,    118,    119,    120,    121,    122, HITS2243
     F   123,    124,    125,    126,    127,    128,    129,    130,    131, HITS2244
     G   132,    133,    134,    135,   -201,   -202,   -204,   -205,   -206/ HITS2245
      DATA  KOE / 70, 150, 150 /                                        HITS2246
      DO 10 J=1,2                                                       HITS2247
      DO 10 I=1,200                                                     HITS2248
   10 IA(I,J) = 0                                                       HITS2249
      DO 20 I=1,200                                                     HITS2250
   20 B(I) = 0.0D0                                                      HITS2251
C FIRST CARD IN TELLS WHICH OF THE 3 IOE DATA STATEMENTS (IY) WILL BE   HITS2252
C   PLACED INTO THE FIRST COLUMN OF IA.                                 HITS2253
      READ (INC,9001,END=70)  IY, IOPRNT, ISPRNT, MCOPT, MCALC, NCELL,  HITS2254
     1 NTRIAL, IRJ1, IRJ2, MCPRNT, IRANNO                               HITS2255
     3,IOPLOT,ISPLOT                                                    HITS2256
 9001 FORMAT(I1, 2I3,10I5)                                              HITS2257
C                                                                       HITS2258
C  INPUT LISTING                                                        HITS2259
      WRITE(IOUT,5000)                                                  HITS2260
 5000 FORMAT(1H1, 43X,'*** SIMULATION INPUT SUMMARY ***'////)           HITS2261
      WRITE(IOUT,5001) IY,IOPRNT,ISPRNT,MCOPT, MCALC,NCELL,NTRIAL,      HITS2262
     AIRJ1,IRJ2,MCPRNT,IRANNO,IOPLOT,ISPLOT                             HITS2263
 5001 FORMAT(30X,                                                       HITS2264
     1         'SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONHITS2265
     ATROLS',////                                                       HITS2266
     4             ,35X,'    IY = ',4X,I1,5X,'IOPRNT = ',2X,I3,5X,'ISPRNT = 'HITS2267
     2,I5,///,50X,                                                      HITS2268
     B              'MONTE CARLO CONTROLS',//,25X,'MCOPT = ',I5,6X,'MCALC =HITS2269
     C ',I5,5X,' NCELL = ',I5,5X,'NTRIAL = ',I5,                        HITS2270
     3                                       //,24X,'  IRJ1 = ',I5,5X,  HITS2271
     D'   IRJ2 = ',I5,5X,'MCPRNT = ',I5,5X,'IRANNO = ',I5,              HITS2272
     E//,24X,'IOPLOT = ',I5,5X,'ISPLOT = ',I5,/////)                    HITS2273
```

```
      WRITE(IOUT,5002)                                                  HITS2274
 5002 FORMAT(1H1,42X,'*** INPUT VARIABLE SPECIFICATIONS ***',///,       HITS2275
     A10X,6X,'CODE',6X,4X,'VARIABLE',4X,5X,'NOMINAL',4X,4X,'TOLERANCE', HITS2276
     B3X,4X,'STANDARD',6X,3X,'SUBSEQUENT',2X,/                          HITS2277
     C10X,5X,'NUMBER',5X,6X,'TYPE',6X,6X,'VALUE',5X,1( ',4X,'DEVIATION', HITS2278
     D5X,5X,'POINTS',5X,//)                                             HITS2279
      CALL RAND ( IRANNO )                                              HITS2280
      DO 25 I=1,20                                                      HITS2281
   25 X = RANDOM ( Y )                                                  HITS2282
C SINCE THE ENTRIES IN THE DATA STATEMENTS ARE PACKED TIGHT, IADD WILL  HITS2283
C    INDICATE HOW MANY TO ADD ON TO THE LOOP COUNTER (IN ORDER TO PICK  HITS2284
C    UP THE CORRECT IOE).                                               HITS2285
      IADD = 0                                                          HITS2286
      IF ( IY .GT. 1 )  IADD = 160                                      HITS2287
      IF ( IY .GT. 2 )  IADD = 310                                      HITS2288
C KA IS THE COUNT OF THE INDEPENDENT VARIABLES IN THE FIRST COLUMN OF IAHITS2289
      KA = KOE(IY)                                                      HITS2290
      DO 30 I=1,KA                                                      HITS2291
   30 IA(I,1) = IOE(I+IADD)                                             HITS2292
      GO TO (40, 50, 60), IY                                            HITS2293
   40 CALL TG2987                                                       HITS2294
      RETURN                                                            HITS2295
   50 CALL TG2440                                                       HITS2296
      RETURN                                                            HITS2297
   60 CALL TG1795                                                       HITS2298
      RETURN                                                            HITS2299
 70   WRITE(IOUT,9002)                                                  HITS2300
 9002 FORMAT(1H1,59(/,1X,6('THATS ALL FOLKS',5X)))                      HITS2301
      CALL EXIT                                                         HITS2302
      RETURN                                                            HITS2303
      END                                                               HITS2304
```

D.31 Subroutine INLHST

```
      SUBROUTINE INLHST                                                 HITS2305
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2306
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS2307
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS2308
C     THIS SUBROUTINE ZEROS OUT THE HISTOGRAM AND MONTE CARLO COUNTERS  HITS2309
      NCC=0                                                             HITS2310
      NR1=0                                                             HITS2311
      NR2=0                                                             HITS2312
      IN8=0                                                             HITS2313
      DO 1 I=1,10                                                       HITS2314
      IR1(I)=0                                                          HITS2315
    1  IR2(I)=0                                                         HITS2316
      DO 2 I=1,200                                                      HITS2317
      ID1(I)=0                                                          HITS2318
    2  ID2(I)=0                                                         HITS2319
      DO 3 I=1,400                                                      HITS2320
      II1(I) = 0                                                        HITS2321
    3 II2(I)=0                                                          HITS2322
      RETURN                                                            HITS2323
      END                                                               HITS2324
```

D.32 Subroutine LOADER

```
      SUBROUTINE LOADER (IV, CN, CV, FCTR, NRR, PM, PV, PT, NCELL          HITS2325
     1 , A , IPASS  )                                                      HITS2326
            IMPLICIT REAL * 8 (A-H , O - Y )                               HITS2327
       DIMENSION  A(20 , 5 )                                               HITS2328
     1 , IB(2,200)                                                         HITS2329
        EQUIVALENCE  (IB(1,1), B(1)  )                                     HITS2330
       COMMON/ HISPLT/ IOPLOT    ,ISPLOT                                   HITS2331
     1                          , ZTITLE(20,30) , ZOATAD(21,4,30)          HITS2332
       COMMON /CARDCM / B(200) , C(5000) , IA(200,2) , ID(10,2) , INC      HITS2333
     1 ,IOPRNT , IOUT , IY ,KARSI(3) , KA , KB , KC , KD , KT1 , K234      HITS2334
     2 , LABC(3)                                                           HITS2335
      DATA    PI / 3.141592654D0  /                                        HITS2336
       DATA   J /0 /                                                       HITS2337
          J= J+1                                                           HITS2338
        IF(  J .GT.  KD+K234)  J= 1                                        HITS2339
          GO TO (  1 , 2  , 3 , 4 ) , IPASS                                HITS2340
 1        JSTRT = 1                                                        HITS2341
            JD = 0                                                         HITS2342
            J2 = 2                                                         HITS2343
            JEND = KD                                                      HITS2344
            JS = 4                                                         HITS2345
           GO TO  5                                                        HITS2346
 2         JEND =  KD +K234                                                HITS2347
           JSTRT  = 1 + KD                                                 HITS2348
       JS=4                                                                HITS2349
           J2 = 2                                                          HITS2350
           JD = 0                                                          HITS2351
          GO TO  5                                                         HITS2352
 3        JSTRT= 1                                                         HITS2353
          JEND = KD                                                        HITS2354
            JS=11                                                          HITS2355
           J2 = 4                                                          HITS2356
           JD = 10                                                         HITS2357
           GO TO 5                                                         HITS2358
 4       JSTRT  =  1 + KD                                                  HITS2359
         JEND   =  KD + K234                                               HITS2360
            JS=11                                                          HITS2361
          J2 =  3                                                          HITS2362
           GO TO  5                                                        HITS2363
 5     FPI = 1.D0 / DSQRT( 2.D0 * PI )                                     HITS2364
          NCELL1 = NCELL +1                                                HITS2365
            IF( J .LE.  KD          ) GO TO 10                             HITS2366
       ITYPE =  IB(1,IA(J-KD,2))                                           HITS2367
         ZTITLE(20, J) =  0.                                               HITS2368
         IF( ITYPE  .EQ.   2  )  ZTITLE(20,J) =  1.                        HITS2369
               GO TO 11                                                    HITS2370
 10     ITYPE = 7                                                          HITS2371
       IF(  ID(J,2) .NE.0 )  ITYPE = 8                                     HITS2372
 11       ZTITLE(1   ,J)=IPASS                                             HITS2373
          ZTITLE(2   ,J)= IV                                               HITS2374
```

D.32 Subroutine LOADER (continued)

```
          ZTITLE(3   ,J)= ITYPE                                       HITS2375
          ZTITLE(JS  ,J)= NRR                                         HITS2376
          ZTITLE(JS+1,J)= CM                                          HITS2377
          ZTITLE(JS+2,J)= CV                                          HITS2378
          ZTITLE(JS+3,J)= PM                                          HITS2379
          ZTITLE(JS+4,J)= PV                                          HITS2380
          ZTITLE(JS+5,J)= FCTR                                        HITS2381
          ZTITLE(JS+6,J)= PT                                          HITS2382
          ZTITLE(18  ,J)= NCELL                                       HITS2383
              SUM = 0.D0                                              HITS2384
             DO   7  IN =  1, NCELL                                   HITS2385
 7           SUM = SUM    +  A(IN,4)                                  HITS2386
             SUM = 100.D0   / SUM                                     HITS2387
            SAVE =  -1.D20                                            HITS2388
             DO  8     IN =   1 , NCELL                               HITS2389
             ZDATAD(IN, J2 ,   J)     = A(IN,4)    * SUM              HITS2390
          IF( SAVE .LT.  ZDATAD(IN,J2,J ) )  SAVE = ZDATAD(IN,J2, J)  HITS2391
             IF( IPASS .EQ. 4 ) GO TO 8                               HITS2392
            IF( IPASS .EQ. 2 ) GO TO 9                                HITS2393
           ZDATAD(IN,  IPASS , J ) = A(IN,2)                          HITS2394
              GO TO  8                                                HITS2395
 9          ZDATAD(IN, 1 ,       J ) = A(IN , 2 )                     HITS2396
         IF( ITYPE  .NE.  2 ) GO TO 20                                HITS2397
          F =   DEXP( - (A(IN,1) - ZTITLE(7,J) )**2 /2.               HITS2398
     1 /ZTITLE(8,J)   ) * FPI /    SQRT( ZTITLE(8,J)   )              HITS2399
         ZTITLE(20,J) = 1                                             HITS2400
          GO TO  21                                                   HITS2401
 20      IF(  ITYPE  .NE.  3 )  GO TO 22                              HITS2402
        ZTITLE(20,J) = 0                                              HITS2403
        F  =  1.D0 / ( ZDATAD( NCELL +1 ,1, J) -ZDATAD(1,1,J)  )      HITS2404
         GO TO 21                                                     HITS2405
 22     IF(  ITYPE .NE.  4   ) GO TO 23                               HITS2406
         KB = IA(J-KD,2)                                              HITS2407
         ITL =  IB(1, KB + 4)                                         HITS2408
         KC  =  IB(1, KB + 5)                                         HITS2409
        DATA=  A(IN,1)                                                HITS2410
        CALL  ARTLU(1 , DATA           , C(KC +ITL),  F, C(KC)  )     HITS2411
 21     ZDATAD(IN,4,J) =  F  *  ZTITLE(9,     J)   *100.D0            HITS2412
 23   CONTINUE                                                        HITS2413
 8      CONTINUE                                                      HITS2414
        IF(  IPASS .NE.  2 )  GO TO 25                                HITS2415
 27       ZDATAD(NCELL1,1,J) = A(NCELL,3)                             HITS2416
          ZDATAD(NCELL1,2,J) = 0.D0                                   HITS2417
C                                                                     HITS2418
C                                                                     HITS2419
C                                                                     HITS2420
              GO TO 26                                                HITS2421
 25     IF( IPASS .EQ. 1 )   GO TO 27                                 HITS2422
        IF( IPASS .EQ. 4 )   GO TO 28                                 HITS2423
C IPASS = 3                                                           HITS2424
```

D.32 Subroutine LOADER (concluded)

```
      ZDATAD(NCELL1, 3 ,J) = A(NCELL , 3)                               HITS2425
      ZDATAD(NCELL1, 4, J) = 0.D0                                       HITS2426
         GO  TO  26                                                     HITS2427
   28 ZDATAD  (NCELL1,3,J)  = 0.D0                                      HITS2428
   26 CONTINUE                                                          HITS2429
       IF ( IPASS .EQ.  1  .OR.  IPASS .EQ. 2 )  GO TO 14               HITS2430
       IF(  ZTITLE(19,J) .LT. SAVE )  ZTITLE(19,J) = SAVE               HITS2431
         GO TO 15                                                       HITS2432
   14      ZTITLE(19,J) = SAVE                                          HITS2433
   15 CONTINUE                                                          HITS2434
C       IF( IPASS .EQ. 4 )                                              HITS2435
C       CALL HISPLT                                                     HITS2436
          RETURN                                                        HITS2437
           END                                                          HITS2438
```

D.33 **Subroutine MCRL**

```
      SUBROUTINE MCRL                                                   HITS2439
      IMPLICIT REAL*8 (A-H,O-Y)                                         HITS2440
      COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS2441
     1IY,KARSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS2442
      COMMON /CMCPRN/ MCPRNT                                            HITS2443
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS2444
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS2445
      COMMON/HISPLT/IOPLOT,ISPLOT,ZTITLE(3120)                          HITS2446
      COMMON/OEC/OE(600)                                                HITS2447
      KCASE = 0                                                         HITS2448
      DO 400 IBB = 1, 3120                                              HITS2449
  400 ZTITLE(IBB) = 0.0                                                 HITS2450
      CALL INLHST                                                       HITS2451
      CALL CHCKIN                                                       HITS2452
      IF(MCPRNT.EQ.1) WRITE(IOUT,5000)                                  HITS2453
 5000 FORMAT(1H1,39X,'*** SUMMARY OF MONTE CARLO RANDOM EXPERIMENTS ***'HITS2454
     A,/////)                                                           HITS2455
C       START MONTE CARLO LOOP WITH SAMPLE GENERATOR                    HITS2456
 1      CALL SAMPLE                                                     HITS2457
C       RUN CASE                                                        HITS2458
      KCASE = KCASE + 1                                                 HITS2459
      IF ( MCALC )  20,20,25                                            HITS2460
   20 CALL PICKI                                                        HITS2461
      GO TO 22                                                          HITS2462
   25 CALL CRVFT (MCALC)                                                HITS2463
   22 IF ( MCPRNT )  24,24,23                                           HITS2464
   23 CONTINUE                                                          HITS2465
C USE THE LAST TEN ENTRIES IN THE C ARRAY FOR OUTPUT.                   HITS2466
      K = 0                                                             HITS2467
  231 JC = 4990                                                         HITS2468
  232 K = K + 1                                                         HITS2469
      IF ( K .GT. K234 )  GO TO 233                                     HITS2470
      JA = IA(K,1)                                                      HITS2471
      JC = JC + 1                                                       HITS2472
      C(JC) = OE(JA)                                                    HITS2473
      IF ( JC .LT. 5000 )  GO TO 232                                    HITS2474
  233 WRITE (IOUT,9002)KCASE,  (C(I), I=4991,JC)                        HITS2475
 9002 FORMAT(1X,'CASE',I5,/,1X,'IND. VAR.',1P10E12.4)                   HITS2476
      IF ( K .LT. K234 )  GO TO 231                                     HITS2477
      K = 0                                                             HITS2478
      JC = 4990                                                         HITS2479
  234 K = K + 1                                                         HITS2480
      IF ( K .GT. KD )  GO TO 235                                       HITS2481
      JD = ID(K,1)                                                      HITS2482
      JC = JC + 1                                                       HITS2483
      C(JC) = OE(JD)                                                    HITS2484
      GO TO 234                                                         HITS2485
  235 WRITE (IOUT,9003)  (C(I), I=4991,JC)                              HITS2486
 9003 FORMAT(1X,'DEP. VAR.',1P10E12.4,///)                              HITS2487
   24 IER = 0                                                           HITS2488
```

D.33 Subroutine MCRL (concluded)

```
C       PERFORM INTERVAL CHECK                                          HITS2489
        IER=0                                                           HITS2490
        CALL HSTG1(IER)                                                 HITS2491
        IF(IER)2,3,3                                                    HITS2492
 2      NR1=NR1+1                                                       HITS2493
        IF(NR1-IRJ1)3,3,7                                               HITS2494
 3      IF(IN8)6,6,4                                                    HITS2495
C       IF THERE ARE TYPE 8,S PERFORM CHECK ON USER SPECS               HITS2496
 4      IER=0                                                           HITS2497
        CALL HSTG2(IER)                                                 HITS2498
        IF(IER)5,6,6                                                    HITS2499
 5      NR2=NR2+1                                                       HITS2500
        IF(NR2-IRJ2)6,6,7                                               HITS2501
C       INCREMENT SAMPLE COUNTER AND CHECK - - END OF MONTE CARLO LOOP  HITS2502
 6      NCC=NCC+1                                                       HITS2503
        IF(NCC-NTRIAL)1,7,7                                             HITS2504
 7      IF(IN8)8,8,9                                                    HITS2505
 8      KEND=2                                                          HITS2506
        GO TO 10                                                        HITS2507
 9      KEND=4                                                          HITS2508
C       START PRINTING RESULTS                                          HITS2509
 10     WRITE(IOUT,200) NTRIAL,NR1,NR2                                  HITS2510
 200    FORMAT(1H1,2X,'MONTE CARLO RESULTS',10X,'SAMPLE =',I5,4X,'INTERNALHITS2511
       1 REJECTS =',I4,4X,'USER REJECTS =',I4)                          HITS2512
        DO 21 L=1,KEND                                                  HITS2513
        GO TO(12,13,14,15),L                                            HITS2514
 12     WRITE(IOUT,201)                                                 HITS2515
  201   FORMAT( // 20X, 'DEPENDENT VARIABLES WITH INTERNAL RANGES')     HITS2516
        GO TO 11                                                        HITS2517
 13     WRITE(IOUT,202)                                                 HITS2518
 202    FORMAT(//20X,'INDEPENDENT VARIABLES WITH INTERNAL RANGES')      HITS2519
        GO TO 11                                                        HITS2520
 14     WRITE(IOUT,203)                                                 HITS2521
 203    FORMAT(//20X,'DEPENDENT VARIABLES WITH USERS RANGES')           HITS2522
        GO TO 11                                                        HITS2523
 15     WRITE(IOUT,204)                                                 HITS2524
 204    FORMAT(//20X,'SORTED INDEPENDENT VARIABLES FOR USERS RANGES')   HITS2525
 11     CALL PRNTMC(L)                                                  HITS2526
   21   CONTINUE                                                        HITS2527
        RETURN                                                          HITS2528
        END                                                             HITS2529
```

.34 Subroutine MOVEUP

```
      SUBROUTINE MOVEUP                                                 HITS2530
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2531
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2),   NC,      HITS2532
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS2533
      DIMENSION  IB(2,200)                                              HITS2534
      EQUIVALENCE  (IB(1,1), B(1))                                      HITS2535
C IF THERE WERE ANY TYPE 1, MOVE THEM UP TO THE START OF THE IA ARRAY.  HITS2536
      IF ( KT1 .EQ. 0 )  GO TO 20                                       HITS2537
      KT1R = 0                                                          HITS2538
   10 KT1R = KT1R + 1                                                   HITS2539
      J1 = KARS1(KT1R)                                                  HITS2540
      JS1 = IA(KT1R,1)                                                  HITS2541
      JS2 = IA(KT1R,2)                                                  HITS2542
      IA(KT1R,1) = IA(J1,1)                                             HITS2543
      IA(KT1R,2) = IA(J1,2)                                             HITS2544
      IA(J1,1) = JS1                                                    HITS2545
      IA(J1,2) = JS2                                                    HITS2546
      IF ( KT1R .LT. KT1 )  GO TO 10                                    HITS2547
      GO TO 50                                                          HITS2548
   20 CONTINUE                                                          HITS2549
      IF ( K234 .EQ. 0 )  GO TO 50                                      HITS2550
C RUN THRU IA.  GET SUBSCRIPT OF B.  IF TYPE 2, 3, OR 4, MOVE UP IN IA. HITS2551
      K234R = 0                                                         HITS2552
      IAR = 0                                                           HITS2553
   30 IAR = IAR + 1                                                     HITS2554
      IF ( IA(IAR,2) .EQ. 0  .OR.  IA(IAR,2) .EQ. -1 )  GO TO 40        HITS2555
      JB = IA(IAR,2)                                                    HITS2556
      IF ( IB(1,JB) .EQ. 1  .OR.  IB(1,JB) .GT. 4 )  GO TO 40           HITS2557
      K234R = K234R + 1                                                 HITS2558
      JS1 = IA(K234R,1)                                                 HITS2559
      JS2 = IA(K234R,2)                                                 HITS2560
      IA(K234R,1) = IA(IAR,1)                                           HITS2561
      IA(K234R,2) = IA(IAR,2)                                           HITS2562
      IA(IAR,1) = JS1                                                   HITS2563
      IA(IAR,2) = JS2                                                   HITS2564
   40 IF ( K234R .EQ. K234 )       GO TO 50                             HITS2565
      IF ( IAR  LT. KA )  GO TO 30                                      HITS2565
      WRITE (IOUT,9006)                                                 HITS2567
 9006 FORMAT(71H MOVEUP ERROR ... REACHED END OF IA BEFORE FINDING ALL THITS2568
     1YPE 2, 3 AND 4,)                                                  HITS2569
      CALL EXIT                                                         HITS2570
   50 CONTINUE                                                          HITS2571
      RETURN                                                            HITS2572
      END                                                               HITS2573
```

D.35 Function NCOV

```
      FUNCTION NCOV(II, JJ)                                             HITS2574
      IMPLICIT REAL*8(A-H, O-Z)                                         HITS2575
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,       HITS2576
     1 IOPRNT, IOUT                                                     HITS2577
       COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV            HITS2578
      IF(II.NE.JJ) GO TO 100                                            HITS2579
      WRITE(IOUT,99) II, JJ                                             HITS2580
   99 FORMAT( ///, ' DO234 TRIES TO LOOK UP VARIANCE IN COVARIANCE ARRAYHITS2581
     A  I = ', I3, ' J = ', I3)                                         HITS2582
  100 CONTINUE                                                          HITS2583
       IF ( JJ.LT. II) GO TO 1                                          HITS2584
      I= II                                                             HITS2585
      J= JJ                                                             HITS2586
      GO TO 2                                                           HITS2587
    1  I= JJ                                                            HITS2588
      J= II                                                             HITS2589
    2   CONTINUE                                                        HITS2590
      NSUM= 0                                                           HITS2591
      I1= I-1                                                           HITS2592
      IF( I1 .EQ. 0) GO TO 3                                            HITS2593
      DO 4 IK= 1, I1                                                    HITS2594
    4  NSUM= NSUM + K234 -IK                                            HITS2595
    3     CONTINUE                                                      HITS2596
       NCOV=  KCOV + NSUM +J -I                                         HITS2597
      IF(NCOV.LE.2000) RETURN                                           HITS2598
      WRITE(IOUT,98) NCOV                                               HITS2599
   98 FORMAT( ///, ' COVARIANCE MATRIX EXCEEDS THE C ARRAY.  NCOV = ',  HITS2600
     A I4)                                                              HITS2601
       RETURN                                                           HITS2602
      END                                                               HITS2603
```

D.36 Function NMEAN

```
      FUNCTION NMEAN(I)                                                 HITS2604
     IMPLICIT REAL*8(A-H, O-Z)                                          HITS2605
    COMMON /CARDCM/  B(200), C(5000), IA(200,2)                         HITS2606
      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV             HITS2607
      NMEAN= KCSUMS+I                                                   HITS2608
      RETURN                                                            HITS2609
      END                                                               HITS2610
```

D.37 Function NVAR

```
      FUNCTION  NVAR(I)                                                 HITS2611
    IMPLICIT REAL*8(A-H, O-Z)                                           HITS2612
    COMMON /CARDCM/  B(200), C(5000), IA(200,2)                         HITS2613
     COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV               HITS2614
     NVAR=  KCSUMS +KD +I                                               HITS2615
     RETURN                                                             HITS2616
      END                                                               HITS2617
```

D.38 Function NVARX

```
      FUNCTION  NVARX(I)                                               HITS2618
     IMPLICIT REAL*8(A-H, O-Z)                                         HITS2619
     COMMON /CARDCM/  B(200), C(5000), IA(200,2)                       HITS2620
      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV            HITS2621
       NVARX=  IA(I,2) +3                                              HITS2622
      RETURN                                                           HITS2623
      END                                                              HITS2624
```

D.39 Function N1D

```
      FUNCTION   N1D(I,J)                                                  HITS2625
      IMPLICIT REAL*8(A-H, O-Z)                                           HITS2626
      COMMON /CARDCM/  B(200), C(5000), IA(200,2)                         HITS2627
       COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV              HITS2628
        N1D=  KCYBAR+ (2*J-1)*KD +I                                       HITS2629
       RETURN                                                             HITS2630
      END                                                                 HITS2631
```

D.40 Function N2D

```
      FUNCTION N2D(I, JJ, KK)                                         HITS2632
      IMPLICIT REAL*8(A-H, O-Z)                                       HITS2633
      COMMON /CARDCM/  B(200), C(5000), IA(200,2)                     HITS2634
      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV            HITS2635
      IF( KK.LT. JJ) GO TO 1                                          HITS2636
      J= JJ                                                           HITS2637
      K= KK                                                           HITS2638
        GO TO 2                                                       HITS2639
 1      J= KK                                                         HITS2640
      K= JJ                                                           HITS2641
 2    CONTINUE                                                        HITS2642
        IF( J.NE, K)  GO TO 3                                         HITS2643
      N2D=  KCYBAR +    2* J *KD +I                                   HITS2644
      RETURN                                                          HITS2645
 3    CONTINUE                                                        HITS2646
      J1= J-1                                                         HITS2647
      NSUM=0                                                          HITS2648
      IF( J1 .EQ. 0) GO TO 5                                          HITS2649
      DO 4 JK= 1, J1                                                  HITS2650
 4      NSUM= NSUM + K234 -JK                                         HITS2651
 5     CONTINUE                                                       HITS2652
       N2D= KCDOUB + KD *(NSUM +K -J-1) +I                            HITS2653
      RETURN                                                          HITS2654
      END                                                             HITS2655
```

D.41 Subroutine PICK1

```
      SUBROUTINE PICK1                                                  HITS2556
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2657
      COMMON /CARDCM/  B(200), C(500), IA(200,2), ID(10,2), INC,        HITS2558
     1 IOPRNT, IOUT, IY, KARSI(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS2659
      GO TO ( 1, 2, 3 ), IY                                             HITS2660
    1 CALL G2987                                                        HITS2661
      RETURN                                                            HITS2662
    2 CALL G2440                                                        HITS2663
      RETURN                                                            HITS2664
    3 CALL G1795                                                        HITS2665
      RETURN                                                            HITS2666
      END                                                               HITS2667
```

```
      SUBROUTINE PRNTMC(IPASS)                                          HITS2668
      IMPLICIT REAL*8 (A-H,O-Y)                                         HITS2669
      COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT, HITS2670
     1IY,KARSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS2671
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS2672
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS2673
      DIMENSION A(20,5)                                                 HITS2674
      COMMON/HISPLT/ IOPLOT,ISPLOT,ZTITLE(20,30),ZDATAD(21,4,30)        HITS2675
      EQUIVALENCE (A(1,1),C(4900))                                      HITS2676
      EQUIVALENCE  (KADD,KC)                                            HITS2677
C     THIS ROUTINE NEEDS THE KADD ADDRESS OF THE NOMINAL CASE RESULTS   HITS2678
      IBN=0                                                             HITS2679
      ENCL=NCELL                                                        HITS2680
      GO TO(1,2,1,2),IPASS                                              HITS2681
    1 KE=KD                                                             HITS2682
      LADD=2*KD*K234 + KD + (KADD - 1)                                  HITS2683
      GO TO 3                                                           HITS2684
    2 KE=K234                                                           HITS2685
    3 DO 17 I=1,KE                                                      HITS2686
      GO TO(4,5,6,5),IPASS                                              HITS2687
C     FIRST PASS FOR THE DEPENDENT VARIABLES                            HITS2688
C     DEFINITIONS  PM - PRIOR MEAN VALUE     PV - PRIOR VARIANCE        HITS2689
C                  PT - PRIOR TOLERANCE    FCTR - CLASS INTERVAL        HITS2690
C                  IV - VARIABLE CODE NO    NRR - NUMBER OF REJECTS     HITS2691
    4 IV=ID(I,1)                                                        HITS2692
      IMN=LADD + I                                                      HITS2693
      IVAR=IMN+KD                                                       HITS2694
      PV=C(IVAR)                                                        HITS2695
      PT=3.*DSQRT(PV)                                                   HITS2696
      PM=C(IMN)                                                         HITS2697
      FCTR = 2.0D0 * PT / ENCL                                          HITS2698
      NRR=IR1(I)                                                        HITS2699
      GO TO 9                                                           HITS2700
C     FIRST AND THIRD PAS FOR INDEPENDENT VARIABLES                     HITS2701
    5 IV=IA(I,1)                                                        HITS2702
      L=IA(I,2)                                                         HITS2703
      NRR=0                                                             HITS2704
    8 PM=B(L+1)                                                         HITS2705
      PT=B(L+2)                                                         HITS2706
      PV=B(L+3)                                                         HITS2707
      FCTR = 2.0D0 * PT / ENCL                                          HITS2708
      GO TO 9                                                           HITS2709
C     FOURTH PASS FOR USER SPEC DEP. VARIABLES                          HITS2710
    6 L=ID(I,2)                                                         HITS2711
C     CHECK IF IT IS A TYPE 8                                           HITS2712
      IF(L)4,4,7                                                        HITS2713
    7 IV=ID(I,1)                                                        HITS2714
      NRR=IR2(I)                                                        HITS2715
      GO TO 8                                                           HITS2716
C     CONVERT AND TRANSFER HISTOGRAMS TO OUTPUT ARRAY A                 HITS2717
```

D.42 Subroutine PRNTMC (concluded)

```
 9    DO 14 J=1,NCELL                                                       HITS2718
      GO TO(10,11,12,13),IPASS                                              HITS2719
10    A(J,4)=ID1(IBN+J)                                                     HITS2720
      GO TO 14                                                              HITS2721
11    A(J,4)=II1(IBN+J)                                                     HITS2722
      GO TO 14                                                              HITS2723
12    A(J,4)=ID2(IBN+J)                                                     HITS2724
      GO TO 14                                                              HITS2725
13    A(J,4)=II2(IBN+J)                                                     HITS2726
14    CONTINUE                                                              HITS2727
C     FILL IN THE OUTPUT ARRAY AND CALC MEAN AND VARIANCE                   HITS2728
      CALL FILLA(A,FCTR,PM,PT,CM,CV,NCELL)                                  HITS2729
      IF(IOPLOT.NE.0)                                                       HITS2730
     1CALL LOADER(IV,CM,CV,FCTR,NRR,PM,PV,PT,NCELL,A,IPASS)                 HITS2731
      WRITE(IOUT,200) IV,CM,CV,FCTR,NRR,PM,PV,PT                            HITS2732
200   FORMAT(///2X,'CODE NUMBER =',I5,4X,'MEAN =',E14.6,4X,'VAR =',E14.6HITS2733
     1,4X,'INTERVAL =',E14.6/2X,'NO. REJECTS =',I5,2X,'(MEAN) =',E14.6, HITS2734
     22X,'(VAR) =',E14.6,3X,'TOLERANCE =',E14.6//4X,'CLASS MARK',3X,    HITS2735
     3'LOWER BOUND',3X,'UPPER BOUND',1X,'FREQ',3X,'DIST. FUNCT.')      HITS2736
      DO 16 K=1,NCELL                                                       HITS2737
16    WRITE(IOUT,201) (A(K,L),L=1,5)                                        HITS2738
201   FORMAT(3E14.6,F5.0,E14.6)                                             HITS2739
17    IBN=IBN+NCELL                                                         HITS2740
      RETURN                                                                HITS2741
      END                                                                   HITS2742
```

D.43 Subroutine QCOR

```
      SUBROUTINE   QCOR
      IMPLICIT REAL*8(A-H, O-Z)
      COMMON /CARDCM/  B(200), C(5000), IA(200,2)
       COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
            K1 = K234 - 1
            IF( K1 .EQ.  0)  RETURN
           DO  1    I =  1 ,  KD
             DEL  = 0.D0
              DELP= 0.D0
          DO    2    J = 1, K1
           NXJ  =  N1D(I,J)
           YXJ  =  C( NXJ)
            KB  = J + 1
          DO    2    K = KB , K234
         N2   =  N2D( I,J,K)
         NK =    N1D(I,K)
        NC   =  NCOV(J,K)
       CV =   C(NC)
         DEL   = DEL +  C( N2)   * CV
    2    DELP = DELP+  YXJ  *   C( NK) * CV
         DELP =   DELP * 2.D0
         NY =  NMEAN(I)
         C(NY) = C(NY)   + DEL
         NV =  NVAR(I)
    1    C(NV) = C(NV) + DELP
           RETURN
             END
```

D.44 Function RANDOM

```
      FUNCTION RANDOM(X)                                                HITS2770
      IMPLICIT REAL*8(A-H,O-Z)                                          HITS2771
       CALL   RANDU(IX,IY, X)                                           HITS2772
     IX= IY                                                             HITS2773
      RANDOM=X                                                          HITS2774
      RETURN                                                            HITS2775
       ENTRY RAND(IX)                                                   HITS2776
      RETURN                                                            HITS2777
     END                                                                HITS2778
```

D.45 Subroutine RANDU

```
      SUBROUTINE RANDU(IX,IY,YFL)                                     HITS2779
      IMPLICIT REAL*8(A-H,O-Z)                                        HITS2780
      IY= IX *65539                                                   HITS2781
      IF( IY) 5,6,6                                                   HITS2782
    5   IY= IY+2147483647 + 1                                         HITS2783
    6   YFL= IY                                                       HITS2784
      YFL= YFL*.4656613D-9                                            HITS2785
      RETURN                                                          HITS2786
      END                                                             HITS2787
```

```
      SUBROUTINE RW2987                                                HITS2788
      IMPLICIT REAL*8 (A-H,O-Z)                                        HITS2789
      COMMON/DEC/DE(600)                                               HITS2790
      EQUIVALENCE             (VC     ,DE( 101)), (VWIND  ,DE( 102)),  HITS2791
     A(PHIWIN ,DE( 103)), (RHO    ,DE( 104)), (CD1    ,DE( 105)),      HITS2792
     B(V1     ,DE( 106)), (CD2    ,DE( 107)), (V2     ,DE( 108)),      HITS2793
     C(XN     ,DE( 109)), (CNA    ,DE( 110)), (SM     ,DE( 111)),      HITS2794
     D(CMQ    ,DE( 112)), (CMPA   ,DE( 113)), (TRIM   ,DE( 114)),      HITS2795
     E(PNITRI ,DE( 115)), (ANGO   ,DE( 116)), (PHIANG ,DE( 117)),      HITS2796
     F(RATE   ,DE( 118)), (PHIR T ,DE( 119)), (GAMO   ,DE( 120)),      HITS2797
     G(PHIGAM ,DE( 121)), (A      ,DE( 122)), (ELL    ,DE( 123)),      HITS2798
     H(D      ,DE( 124)), (W      ,DE( 125)), (AIX    ,DE( 126)),      HITS2799
     I(AIY    ,DE( 127)), (P      ,DE( 128)), (ALPCON ,DE( 129)),      HITS2800
     J(CAA    ,DE( 130))                                               HITS2801
      EQUIVALENCE             (ALPHA  ,DE( 400)), (ATOT   ,DE( 401)),  HITS2802
     A(ALPTRM ,DE( 402)), (ALPHDO ,DE( 403)), (B      ,DE( 404)),      HITS2803
     B(BETTRM ,DE( 405)), (BETAO  ,DE( 406)), (BETADO ,DE( 407)),      HITS2804
     C(BETA   ,DE( 408)), (ALPMAX ,DE( 409)), (ALPMIN ,DE( 410)),      HITS2805
     D(CAYD   ,DE( 411)), (CDB    ,DE( 412)), (CMA    ,DE( 413)),      HITS2806
     E(CMTHTD ,DE( 414)), (CMTHTA ,DE( 415)), (CAY1   ,DE( 416)),      HITS2807
     F(CAY2   ,DE( 417)), (CDAOB  ,DE( 418)), (CDAB   ,DE( 419)),      HITS2808
     G(DELTT  ,DE( 420)), (DELV   ,DE( 421)), (DELT   ,DE( 422)),      HITS2809
     H(DELX   ,DE( 423)), (DYDXO  ,DE( 424)), (DZDXO  ,DE( 425)),      HITS2810
     I(DYDT   ,DE( 426)), (DZDTO  ,DE( 427)), (DELW   ,DE( 428)),      HITS2811
     J(DELLAM ,DE( 429)), (EYEP   ,DE( 430)), (EMP    ,DE( 431)),      HITS2812
     K(EALT   ,DE( 432)), (EDLT   ,DE( 433)), (F      ,DE( 434)),      HITS2813
     L(H      ,DE( 435)), (PSIDO  ,DE( 436)), (PHIO   ,DE( 437)),      HITS2814
     M(PHI1   ,DE( 438)), (PHI2   ,DE( 439)), (PSIO   ,DE( 440)),      HITS2815
     N(R1     ,DE( 441)), (R2     ,DE( 442)), (R3     ,DE( 443)),      HITS2816
     O(R4     ,DE( 444)), (RTRIM  ,DE( 445)), (TC     ,DE( 446)),      HITS2817
     P(TCO    ,DE( 447)), (TS     ,DE( 448)), (TGO    ,DE( 449)),      HITS2818
     Q(TP     ,DE( 450)), (THETAO ,DE( 451)), (THETDO ,DE( 452)),      HITS2819
     R(TOL    ,DE( 453)), (VC     ,DE( 454)), (VCO    ,DE( 455)),      HITS2820
     S(VYA    ,DE( 456)), (VZA    ,DE( 457)), (WX     ,DE( 458))       HITS2821
      EQUIVALENCE             (WZ     ,DE( 459)), (WO     ,DE( 460)),  HITS2822
     A(W1     ,DE( 461)), (W2     ,DE( 462)), (WL2    ,DE( 463)),      HITS2823
     B(XLAM   ,DE( 464)), (XLAM1  ,DE( 465)), (XLAM2  ,DE( 466)),      HITS2824
     C(XNU1   ,DE( 467)), (XNU2   ,DE( 468)), (XG     ,DE( 469)),      HITS2825
     D(XMU    ,DE( 470)), (XJAY   ,DE( 471)), (XJAZ   ,DE( 472)),      HITS2826
     E(XJA    ,DE( 473)), (YA     ,DE( 474)), (ZA     ,DE( 475)),      HITS2827
     F(ALPH   ,DE( 476))                                               HITS2828
      EQUIVALENCE             (TN     ,DE( 500)), (XTFC   ,DE( 501)),  HITS2829
     A(YTFC   ,DE( 502)), (ZTFC   ,DE( 503)), (VXTFC  ,DE( 504)),      HITS2830
     B(VYTFC  ,DE( 505)), (VZTFC  ,DE( 506))                           HITS2831
      EQUIVALENCE             (XT     ,DE( 507)), (YT     ,DE( 508)),  HITS2832
     A(ZT     ,DE( 509)), (VXT    ,DE( 510)), (VYT    ,DE( 511)),      HITS2833
     B(VZT    ,DE( 512)), (YR     ,DE( 513)), (ZR     ,DE( 514)),      HITS2834
     C(VXR    ,DE( 515)), (VYR    ,DE( 516)), (VZR    ,DE( 517)),      HITS2835
     D(TX     ,DE( 518))                                               HITS2836
      CALL INCON                                                       HITS2837
```

D.46 Subroutine RW2987 (concluded)

```
      A57 = DLPCON/57.29577951D0                                        HITS2838
      CALL XXC(XLAMO,XC,VO,DELLAM,CAY1,CAY2,TOL,A57)                    HITS2839
      CALL TRAJT(TN,XC,VXT,XT)                                          HITS2840
      CALL CROSS(XT,VXT)                                                HITS2841
      CALL WIND(VXT,TN,XT,WZ,WX,VO,VZW,ZW)                              HITS2842
      YT = YA                                                           HITS2843
      ZT = ZA + ZW                                                      HITS2844
      VYT = VYA                                                         HITS2845
      VZT = VZA + VZW                                                   HITS2846
      CALL TRAJX(XN,XC,VXR,TX)                                          HITS2847
      CALL CROSS(XN,VXR)                                                HITS2848
      CALL WIND(VXR,TX,XN,WZ,WX,VO,VZW,ZW)                              HITS2849
      YR = YA                                                           HITS2850
      ZR = ZA + ZW                                                      HITS2851
      VYR = VYA                                                         HITS2852
      VZR = VZA + VZW                                                   HITS2853
      RETURN                                                            HITS2854
      END                                                               HITS2855
```

D.47 Subroutine SAMPLE

```
      SUBROUTINE SAMPLE                                                 HITS2856
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2857
      COMMON/CARDCM/ B(2000),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,HITS2858
     1IY,KARS1(3),KA,KB,KC,KD,KT1,K234,LABC(3)                          HITS2859
      COMMON/OEC/OE(600)                                                HITS2860
      COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),  HITS2861
     1IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2HITS2862
      DIMENSION IB(2,200)                                               HITS2863
      EQUIVALENCE (IB(1,1),B(1))                                        HITS2864
C       THIS ROUTINE GENERATES THE INPUT SAMPLES AND REQUIRES A UNIFORM  HITS2865
C       RANDOM NUMBER GENERATOR CALLED RANDOM.                           HITS2866
      IBN=1                                                             HITS2867
      ENN=NCELL                                                         HITS2868
      DO 12 I=1,K234                                                    HITS2869
      ILP=0                                                             HITS2870
      IADD=IA(I,2)                                                      HITS2871
 10    CALL RANDOM(RN)                                                  HITS2872
      IF(IB(1,IADD)-3)1,2,2                                             HITS2873
C       IF TYPE NO. IS 2 CONVERT RN TO A GAUSSIAN DISTRIBUTION           HITS2874
  1    CALL CNVRT(RN)                                                   HITS2875
  2   VAL=B(IADD+1)+(2.*RN-1.)*B(IADD+2)                                HITS2876
      RNS=ENN*RN                                                        HITS2877
      INDX=RNS                                                          HITS2878
      IIT(I)=IBN+INDX                                                   HITS2879
C       VAL IS THE RANDOM VALUE AND IIT IS THE HISTOGRAM ADDRESS         HITS2880
      IF(IB(1,IADD)-3)5,5,4                                             HITS2881
C       STORE THE RANDOM VALUE IN THE OE ARRAY                           HITS2882
  5    IADD=IA(I,1)                                                     HITS2883
      OE(IADD)=VAL                                                      HITS2884
      GO TO 12                                                          HITS2885
C       FOR TYPE 4 CALCULATE ACTUAL PRUB. DENSITY FOR CHOSEN RANDOM VALUEHITS2886
  4    INDX=IADD+4                                                      HITS2887
      JY=IB(1,INDX)                                                     HITS2888
      FEN=JY                                                            HITS2889
      FEN=FEN*RN                                                        HITS2890
      J=FEN                                                             HITS2891
      INDX=INDX+1                                                       HITS2892
      JF=IB(1,INDX)+J                                                   HITS2893
      JY=JY+JF                                                          HITS2894
C       JF AND JY ARE ADDRESSES IN THE C ARRAY OF DIST. FUNC. AND VARIABLHITS2895
C       VALUE RESP.--THAT IS VAL IS GT C (JY) AND LT C(JY+1)--INTERPOLATEHITS2896
C       TO DETERMINE THE ACTUAL VALUE OF DIST. FCN. FOR CHOSEN VAL       HITS2897
      FACT=C(JF)+(VAL-C(JY))*(C(JF+1)-C(JF))/(C(JY+1)-C(JY))            HITS2898
C       CALCULATE RANDOM PROBABILITY                                     HITS2899
      CALL RANDOM(RN)                                                   HITS2900
      FTRY=RN*B(IADD+3)                                                 HITS2901
      IF(FACT-FTRY)6,5,5                                                HITS2902
C       IF THE RANDOM VALUE IS GREATER THAN THE ACTUAL REJECT THIS SAMPLEHITS2903
  6    ILP=ILP+1                                                        HITS2904
      IF(ILP-20)10,10,11                                                HITS2905
```

D.47 Subroutine SAMPLE (concluded)

```
C        IF A VALUE CANT BE FOUND IN 20 TRYS QUIT                        HITS2906
 11      WRITE(IOUT,200)                                                 HITS2907
 200    FORMAT(10X,'TYPE 4 RANDOM SAMPLE ERROR')                         HITS2908
        CALL EXIT                                                        HITS2909
 12     IBN=IBN+NCELL                                                    HITS2910
        RETURN                                                           HITS2911
        END                                                              HITS2912
```

```
      SUBROUTINE SPRNT                                                  HITS2913
      IMPLICIT  REAL*8 (A-H, O-Z)                                       HITS2914
      COMMON/NAMES/HEAD(180)                                            HITS2915
      DIMENSION UNITS1(132),UNITS2(48), UNITS(180)                      HITS2916
      EQUIVALENCE (UNITS1(1), UNITS(1)), (UNITS2(1), UNITS(133))        HITS2917
      COMMON/OEC/OE(600)                                                HITS2918
      DIMENSION IOE(2,600)                                              HITS2919
      EQUIVALENCE (IOE(1,1),OE(1))                                      HITS2920
C                                                                       HITS2921
      COMMON /CARDCM/ BB(200),CC(5000), IA(200,2), ID(10,2), INC,       HITS2922
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS2923
      DATA IFOR1/30/,IFOR2/5/,IFOR3/6/,IFOR4/43/,IFOR5/77/,IFOR6/19/    HITS2924
      DATA UNITS1       /          'FT/SEC','FT/SEC','FT/SEC','SL/CFT', HITS2925
     A'(NONE)','FT/SEC','(NONE)','FT/SEC','FT    ','1/RAD ','100/L ',   HITS2926
     B'(NONE)','(NONE)','DEG   ','DEG   ','DEG   ','DEG   ','RAD/SC',   HITS2927
     C'RAD/SC','DEG   ','DEG   ','SQ-FT ','FT    ','FT    ','LBS   ',   HITS2928
     D'SL-SFT','SL-SFT','RAD/SC','DEG   ','1/SDEG',                     HITS2929
     E'FT    ','FT    ','FT    ','FT    ','FT    ',                     HITS2930
     F'(NONE)','(NONE)','(NONE)','(NONE)','(NONE)','(NONE)',            HITS2931
     G'FT    ','FT    ','FT    ','FT    ','FT    ','FT    ','FT    ',   HITS2932
     H'FT    ','FT    ','FT    ','FT/SEC','FT/SEC','FT/SEC','FT/SEC',   HITS2933
     I'FT/SEC','FT/SEC','SEC   ','FT    ','FT    ','FT    ','FT    ',   HITS2934
     J'FT    ','FT    ','FT    ','FT    ','FT    ','FT    ','FT    ',   HITS2935
     K'FT    ','FT    ','FT    ','SQ-FT ','SQ-FT ','SQ-FT ','SQ-FT ',   HITS2936
     L'SQ-FT ','SQ-FT ','SQ-FT ','SQ-FT ','SQ-FT ','SQ-FT ','SQ-FT ',   HITS2937
     M'SQ-FT ','RAD   ','RAD   ','RAD   ','RAD/SC','(NONE)','RAD   ',   HITS2938
     N'RAD   ','RAD/SC','RAD   ','RAD   ','RAD   ','SQ-F/S','(NONE)',   HITS2939
     O'1/RAD ','SEC   ','SEC   ','RAD   ','RAD   ','(NONE)','(NONE)',   HITS2940
     P'SEC   ','FT/SEC','SEC   ','FT    ','RAD   ','RAD   ','FT/SEC',   HITS2941
     Q'FT/SEC','1/SEC ','1/SEC ','SQ-SEC','SEC   ','(NONE)','(NONE)',   HITS2942
     R'1/FT  ','FT/SC2','RAD/SC','RAD   ','RAD   ','RAD   ','RAD   ',   HITS2943
     S'RAD   ','RAD   ','RAD   ','RAD   ','(NONE)','SEC   ','SEC   '/   HITS2944
      DATA UNITS2       /'SEC   ','SEC   ','SEC   ','RAD   ','RAD/SC',   HITS2945
     A'(NONE)','FT/SEC','FT/SEC','FT/SEC','FT/SEC','FT/SEC','FT/SEC',   HITS2946
     B'1/SEC ','1/SEC ','1/SEC ','1/SQ-S','1/SEC ','1/SEC ','1/SEC ',   HITS2947
     C'RAD   ','RAD   ','FT    ','(NONE)','RAD   ','RAD   ','RAD   ',   HITS2948
     D'FT    ','FT    ','RAD   ',                                       HITS2949
     E'SEC   ','FT    ','FT    ','FT    ','FT/SEC','FT/SEC','FT/SEC',   HITS2950
     F'FT    ','FT    ','FT    ','FT/SEC','FT/SEC','FT/SEC','FT    ',   HITS2951
     G'FT    ','FT/SEC','FT/SEC','FT/SEC','SEC   '/                     HITS2952
      WRITE(IOUT,500)                                                   HITS2953
      WRITE(IOUT,501)                                                   HITS2954
      WRITE(IOUT,505)(HEAD(I),OE(I),UNITS(I),I=1,IFOR1)                 HITS2955
      IFOR = IFOR1 + IFOR2                                              HITS2956
      WRITE(IOUT,502)                                                   HITS2957
      WRITE(IOUT,505)(HEAD(I),OE(I+100),UNITS(I),I=1,IFOR)              HITS2958
      WRITE(IOUT,503)                                                   HITS2959
      WRITE(IOUT,506)(HEAD(I+IFOR),IOE(1,I+200),UNITS(I+IFOR),I=1,IFOR3)HITS2960
      IFOR = IFOR + IFOR3                                               HITS2961
      WRITE(IOUT,504)                                                   HITS2962
```

D.48 Subroutine SPRNT (concluded)

```
      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+300),UNITS(I+IFOR),I=1,IFOR4)    HITS2963
  500 FORMAT(1H1,38X,42H*** PROJECTILE DISPERSION CASE SUMMARY ***)      HITS2964
  501 FORMAT(1H0,48X,23HFIRE CONTROL PARAMETERS)                         HITS2965
  502 FORMAT(1H0,49X,21HREAL WORLD PARAMETERS)                           HITS2966
  503 FORMAT(1H0,51X,16HSYSTEMS CONTROLS)                                HITS2967
      IFOR = IFOR + IFOR4                                                HITS2968
      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+399),UNITS(I+IFOR),I=1,IFOR5)    HITS2969
      IFOR = IFOR + IFOR5                                                HITS2970
      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+499),UNITS(I+IFOR),I = 1,IFOR6)  HITS2971
  504 FORMAT(1H ,49X,19HCOMPUTED QUANTITIES)                             HITS2972
  505 FORMAT(     4(2X,A6,3H = ,1PD11.4,1X,A6,2X))                       HITS2973
  506 FORMAT(     4(2X,A6,3H = ,I11   ,1X,A6,2X))                        HITS2974
      RETURN                                                             HITS2975
      END                                                                HITS2976
```

D.49 Subroutine STDDEV

```
      SUBROUTINE STDDEV(A,K,KD,IOUT)                                    HITS2977
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2978
      DIMENSION B(10)                                                   HITS2979
      B(K) = DSQRT(A)                                                   HITS2980
      IF(K.NE.KD) RETURN                                                HITS2981
      WRITE(IOUT,8887)(B(I), I = 1,KD)                                  HITS2982
      WRITE(IOUT,8889)                                                  HITS2983
      WRITE(IOUT,8888)(B(I),B(I),I=1,KD)                                HITS2984
 8887 FORMAT(1H0,9HSTD. DEV.,1X,1P10E12.4)                              HITS2985
 8889 FORMAT(1H0,////,3X,'STANDARD DEVIATIONS WITH HIGHER PRECISION',/) HITS2986
 8888 FORMAT(1H ,1PE12.4,5X,D25.16)                                     HITS2987
      RETURN                                                            HITS2988
      END                                                               HITS2989
```

D.50 Subroutine STOREC

```
      SUBROUTINE STOREC                                                 HITS2990
      IMPLICIT REAL*8 (A-H,O-Z)                                         HITS2991
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,       HITS2992
     1 IOPRNT, IOUT, IY, KARSI(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS2993
      COMMON/OEC/OE(600)                                                HITS2994
C PLACE THE KD DEPENDENT VARIABLES INTO THE C ARRAY.                     HITS2995
      DO 10 I=1,KD                                                      HITS2996
      JOE = ID(I,1)                                                     HITS2997
      KC = KC + 1                                                       HITS2998
      IF(KC GT.5000) WRITE(IOUT,9000) KC                                HITS2999
 9000 FORMAT(10X,'KC = ',I5,1X,'IN STOREC WHICH EXCEEDS THE C ARRAY')   HITS3000
      C(KC) = OE(JOE)                                                   HITS3001
   10 CONTINUE                                                          HITS3002
      RETURN                                                            HITS3003
      END                                                               HITS3004
```

D.51 Subroutine TG1795

```
      SUBROUTINE  TG1795                                              HITS3005
      RETURN                                                          HITS3006
      END                                                             HITS3007
```

D.52 Subroutine TG2440

```
      SUBROUTINE  TG2440                                               HITS3008
      RETURN                                                           HITS3009
      END                                                              HITS3010
```

D.53 Subroutine TG2987

```
      SUBROUTINE TG2987                                                  HITS3011
      IMPLICIT REAL*8 (A-H,O-Z)                                          HITS3012
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,        HITS3013
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)   HITS3014
      COMMON/OEC/OE(600)                                                 HITS3015
      EQUIVALENCE            (VOF     ,OE(    1)), (WXF     ,OE(    2)), HITS3016
     A(WYF     ,OE(    3)), (RHOF    ,OE(    4)), (CD1F    ,OE(    5)), HITS3017
     B(V1F     ,OE(    6)), (CD2F    ,OE(    7)), (V2F     ,OE(    8)), HITS3018
     C(XNF     ,OE(    9)), (CNAF    ,OE(   10)), (SMF     ,OE(   11)), HITS3019
     D(CMQF    ,OE(   12)), (CMPAF   ,OE(   13)), (ATRMSF  ,OE(   14)), HITS3020
     E(BTRMSF  ,OE(   15)), (THETOF  ,OE(   16)), (PSIOF   ,OE(   17)), HITS3021
     F(TDOTOF  ,OE(   18)), (PDOTOF  ,OE(   19)), (GAMOF   ,OE(   20)), HITS3022
     G(AZOF    ,OE(   21)), (AF      ,OE(   22)), (ELLF    ,OE(   23)), HITS3023
     H(DF      ,OE(   24)), (WF      ,OE(   25)), (AIXF    ,OE(   26)), HITS3024
     I(AIYF    ,OE(   27)), (PF      ,OE(   28)), (ALPCOF  ,OE(   29)), HITS3025
     J(CAAF    ,OE(   30))                                               HITS3026
      EQUIVALENCE            (VO      ,OE(  101)), (WX      ,OE(  102)), HITS3027
     A(WY      ,OE(  103)), (RHO     ,OE(  104)), (CD1     ,OE(  105)), HITS3028
     B(V1      ,OE(  106)), (CD2     ,OE(  107)), (V2      ,OE(  108)), HITS3029
     C(XN      ,OE(  109)), (CNA     ,OE(  110)), (SM      ,OE(  111)), HITS3030
     D(CMQ     ,OE(  112)), (CMPA    ,OE(  113)), (ATRMS   ,OE(  114)), HITS3031
     E(BTRMS   ,OE(  115)), (THETO   ,OE(  116)), (PSIO    ,OE(  117)), HITS3032
     F(TDOTO   ,OE(  118)), (FDOTO   ,OE(  119)), (GAMO    ,OE(  120)), HITS3033
     G(AZO     ,OE(  121)), (A       ,OE(  122)), (ELL     ,OE(  123)), HITS3034
     H(D       ,OE(  124)), (W       ,OE(  125)), (AIX     ,OE(  126)), HITS3035
     I(AIY     ,OE(  127)), (P       ,OE(  128)), (ALPCON  ,OE(  129)), HITS3036
     J(CAA     ,OE(  130))                                               HITS3037
      EQUIVALENCE            (XBART   ,OE(  131)), (YBART   ,OE(  132)), HITS3038
     A(ZBART   ,OE(  133)), (YBARX   ,OE(  134)), (ZBARX   ,OE(  135))  HITS3039
      EQUIVALENCE            (IFCRW   ,OE(  201)), (IFCRC   ,OE(  202)), HITS3040
     A(IFC     ,OE(  203))                                               HITS3041
      EQUIVALENCE            (IDUMP   ,OE(  204))                        HITS3042
      EQUIVALENCE            (ICONV   ,OE(  205)), (ICNCL   ,OE(  206))  HITS3043
      EQUIVALENCE            (XBARTI  ,OE(  301)), (YBARTI  ,OE(  302)), HITS3044
     A(ZBARTI  ,OE(  303)), (YBARXI  ,OE(  304)), (ZBARXI  ,OE(  305))  HITS3045
      EQUIVALENCE            (TOL     ,OE(  453))                        HITS3046
C                                                                        HITS3047
C                                                                        HITS3048
C***********************************************************************HITS3049
C                                                                        HITS3050
C    FIRE CONTROL SYSTEM PARAMETERS                                      HITS3051
C                                                                        HITS3052
      VOF     = +1.100000D4                                              HITS3053
      WXF     = +0.000000D0                                              HITS3054
      WYF     = +0.000000D0                                              HITS3055
      RHOF    = +0.002378D0                                              HITS3056
      CD1F    = +3.585398D-2                                             HITS3057
      V1F     = +1.674000D4                                              HITS3058
      CD2F    = +1.195133D-1                                             HITS3059
      V2F     = +3.906000D3                                              HITS3060
```

D.53 Subroutine TG2987 (continued)

```
      XNF    = +1.000000D4                                           HITS3061
      CNAF   = +1.976700D0                                           HITS3062
      SMF    = +6.200000D-2                                          HITS3063
      CMQF   = -7.500000D0                                           HITS3064
      CMPAF  = +0.000000D0                                           HITS3065
      ATRMSF = +0.000000D0                                           HITS3066
      BTRMSF = +0.000000D0                                           HITS3067
      THETOF = +0.000000D0                                           HITS3068
      PSIOF  = +0.000000D0                                           HITS3069
      TDOTOF = +0.000000D0                                           HITS3070
      PDOTOF = +0.000000D0                                           HITS3071
      GAMOF  = +0.000000D0                                           HITS3072
      AZOF   = +0.000000D0                                           HITS3073
      AF     = +3.068000D-3                                          HITS3074
      ELLF   = +3.100000D-1                                          HITS3075
      DF     = +6.250000D-2                                          HITS3076
      WF     = +1.100000D-1                                          HITS3077
      AIXF   = +1.234600D-6                                          HITS3078
      AIYF   = +2.110300D-5                                          HITS3079
      PF     = +4.000000D2                                           HITS3080
      ALPCOF = +5.000000D-1                                          HITS3081
      CAAF   = +0.000000D0                                           HITS3082
C                                                                    HITS3083
C                                                                    HITS3084
C********************************************************************HITS3085
C                                                                    HITS3086
C     PROJECTILE PARAMETERS (REAL WORLD)                             HITS3087
C                                                                    HITS3088
      VO     = +1.100000D4                                           HITS3089
      WX     = +0.000000D0                                           HITS3090
      WY     = +0.000000D0                                           HITS3091
      RHO    = +0.002378D0                                           HITS3092
      CD1    = +3.585398D-2                                          HITS3093
      V1     = +1.674000D4                                           HITS3094
      CD2    = +1.195133D-1                                          HITS3095
      V2     = +3.906000D3                                           HITS3096
      XN     = +1.000000D4                                           HITS3097
      CNA    = +1.976700D0                                           HITS3098
      SM     = +6.200000D-2                                          HITS3099
      CMQ    = -7.500000D0                                           HITS3100
      CMPA   = +0.000000D0                                           HITS3101
      ATRMS  = +0.000000D0                                           HITS3102
      BTRMS  = +0.000000D0                                           HITS3103
      THETO  = +0.000000D0                                           HITS3104
      PSIO   = +0.000000D0                                           HITS3105
      TDOTO  = +0.000000D0                                           HITS3106
      PDOTO  = +0.000000D0                                           HITS3107
      GAMO   = +0.000000D0                                           HITS3108
      AZO    = +0.000000D0                                           HITS3109
      A      = +3.068000D-3                                          HITS3110
```

D-53 Subroutine TG2937 (concluded)

```
      ELL    = +3.1000000D-1                                                HITS3111
      D      = +6.2500000D-2                                                HITS3112
      W      = +1.1000000D-1                                                HITS3113
      AIX    = +1.2346000D-6                                                HITS3114
      AIY    = +2.1103000D-5                                                HITS3115
      P      = +4.0000000D2                                                 HITS3116
      ALPCON = +5.0000000D-1                                                HITS3117
      CAA    = +0.0000000D0                                                 HITS3118
      XBART  = +0.0000000D0                                                 HITS3119
      YBART  = +0.0000000D0                                                 HITS3120
      ZBART  = +0.0000000D0                                                 HITS3121
      YBARX  = +0.0000000D0                                                 HITS3122
      ZBARX  = +0.0000000D0                                                 HITS3123
C                                                                           HITS3124
C                                                                           HITS3125
C**************************************************************************HITS3126
C                                                                           HITS3127
C     SYSTEM CONTROLS                                                       HITS3128
C                                                                           HITS3129
      IFCRW  = 0                                                            HITS3130
      IFCRC  = 0                                                            HITS3131
      IFC    = 1                                                            HITS3132
      IDUMP  = 0                                                            HITS3133
      ICONV  = -10                                                          HITS3134
      ICNCL = 1                                                             HITS3135
C                                                                           HITS3136
C**************************************************************************HITS3137
C                                                                           HITS3138
C     COMPUTED QUANTITIES                                                   HITS3139
C                                                                           HITS3140
      DO 1 I = 301,600                                                      HITS3141
 1    DE(I)  = +0.0000000D0                                                 HITS3142
      TOL = 10.**ICONV                                                      HITS3143
      RETURN                                                                HITS3144
      END                                                                   HITS3145
```

D.54 Subroutine TRAJT

```
      SUBROUTINE  TRAJT( T , XC , VX ,  X  )                            HITS3146
         IMPLICIT REAL * 8 (A-H, O -Z )                                 HITS3147
  COMMON/GEC/OE(600)                                                    HITS3148
  EQUIVALENCE           (V0       ,OE( 101)), (VWIND   ,OE( 102)),      HITS3149
 A(PHIWIN ,OE( 103)), (RHO      ,OE( 104)), (CD1     ,OE( 105)),        HITS3150
 B(V1     ,OE( 106)), (CD2      ,OE( 107)), (V2      ,OE( 108)),        HITS3151
 C(XN     ,OE( 109)), (CNA      ,OE( 110)), (SM      ,OE( 111)),        HITS3152
 D(CMQ    ,OE( 112)), (CMPA     ,OE( 113)), (TRIM    ,OE( 114)),        HITS3153
 E(PNITSI ,OE( 115)), (ANGO     ,OE( 116)), (PHIANG  ,OE( 117)),        HITS3154
 F(RATE   ,OE( 118)), (PHIRAT   ,OE( 119)), (GAMO    ,OE( 120)),        HITS3155
 G(PHIGAM ,OE( 121)), (A        ,OE( 122)), (ELL     ,OE( 123)),        HITS3156
 H(D      ,OE( 124)), (W        ,OE( 125)), (AIX     ,OE( 126)),        HITS3157
 I(AIY    ,OE( 127)), (P        ,OE( 128)), (ALPCON  ,OE( 129)),        HITS3158
 J(CAA    ,OE( 130))                                                    HITS3159
  EQUIVALENCE           (IDUMP    ,OE( 204))                            HITS3160
  EQUIVALENCE           (ALPHA    ,OE( 400)), (ATOT    ,OE( 401)),      HITS3161
 A(ALPTRM ,OE( 402)), (ALPHDO   ,OE( 403)), (B       ,OE( 404)),        HITS3162
 B(BETTRM ,OE( 405)), (BETAO    ,OE( 406)), (BETADO  ,OE( 407)),        HITS3163
 C(BETA   ,OE( 408)), (ALPMAX   ,OE( 409)), (ALPMIN  ,OE( 410)),        HITS3164
 D(CAYD   ,OE( 411)), (CD8      ,OE( 412)), (CMA     ,OE( 413)),        HITS3165
 E(CMTHTD ,OE( 414)), (CMTHTA   ,OE( 415)), (CAY1    ,OE( 416)),        HITS3166
 F(CAY2   ,OE( 417)), (CDAOB    ,OE( 418)), (CDAB    ,OE( 419)),        HITS3167
 G(DELTT  ,OE( 420)), (DELV     ,OE( 421)), (DELT    ,OE( 422)),        HITS3168
 H(DELX   ,OE( 423)), (DYDXO    ,OE( 424)), (DZDXO   ,OE( 425)),        HITS3169
 I(DYDT   ,OE( 426)), (DZDTO    ,OE( 427)), (DELW    ,OE( 428)),        HITS3170
 J(DELLAM ,OE( 429)), (EYEP     ,OE( 430)), (EMP     ,OE( 431)),        HITS3171
 K(EOLT   ,OE( 432)), (EDLT     ,OE( 433)), (F       ,OE( 434)),        HITS3172
 L(H      ,OE( 435)), (PSIDO    ,OE( 436)), (PHIO    ,OE( 437)),        HITS3173
 M(PHI1   ,OE( 438)), (PHI2     ,OE( 439)), (PSIO    ,OE( 440)),        HITS3174
 N(R1     ,OE( 441)), (R2       ,OE( 442)), (R3      ,OE( 443)),        HITS3175
 O(R4     ,OE( 444)), (RTRIM    ,OE( 445)), (TC      ,OE( 446)),        HITS3176
 P(TCO    ,OE( 447)), (TS       ,OE( 448)), (TGO     ,OE( 449)),        HITS3177
 Q(TP     ,OE( 450)), (THETAO   ,OE( 451)), (THETDO  ,OE( 452)),        HITS3178
 R(TOL    ,OE( 453)), (VC       ,OE( 454)), (VCO     ,OE( 455)),        HITS3179
 S(VYA    ,OE( 456)), (VZA      ,OE( 457)), (WX      ,OE( 458))         HITS3180
  EQUIVALENCE           (WZ       ,OE( 459)), (WO      ,OE( 460)),      HITS3181
 A(W1     ,OE( 461)), (W2       ,OE( 462)), (WL2     ,OE( 463)),        HITS3182
 B(XLAM   ,OE( 464)), (XLAM1    ,OE( 465)), (XLAM2   ,OE( 466)),        HITS3183
 C(XNU1   ,OE( 467)), (XNU2     ,OE( 468)), (XG      ,OE( 469)),        HITS3184
 D(XMU    ,OE( 470)), (XJAY     ,OE( 471)), (XJAZ    ,OE( 472)),        HITS3185
 E(XJA    ,OE( 473)), (YA       ,OE( 474)), (ZA      ,OE( 475)),        HITS3186
 F(ALPH   ,OE( 476))                                                    HITS3187
       CALL   VXT(VCO, TCO, VO , WX,F,B , CD8        ,XC,CAVD , TOL )   HITS3188
     IF( DABS( XC) .EQ. 1.D-10)   VCO=VO                                HITS3189
       A2X = A2( XC )                                                   HITS3190
      CDAOB=  CD8    +  DLOG( (CD8 + CAYD/ VCO/VCO                      HITS3191
 A                                       ) / (CD8 + CAYD/VO/VO  )HITS3192
 1  ) / 2.D0 / F / XC                                                   HITS3193
     CDAB = CDAOB +(CNA + CAA) /XC * A2X                                HITS3194
      EF =  DEXP( F   * CDAB* XC)                                       HITS3195
```

```
         EFO=  DEXP( F   * CDAOB * XC )                                   HITS3196
         DELV = VO *(1.DO/ EF - 1.DO / EFO   )                            HITS3197
        DELTT = ((EF -1.DO)/ CDAB  -(EFO -1.DO)/ CDAOB )/ F /VO           HITS3198
          TC =   TCO  +  DELTT                                            HITS3199
          TS  =   T -TC                                                   HITS3200
      IF(IDUMP   .EQ.1)WRITE(6,1004) DELV,DELTT , TC , TS                 HITS3201
 1004  FORMAT(    '      DELV          DELTT          TC            TS'   HITS3202
     1      /    1X, 1P11E12.4   )                                        HITS3203
         IF(     DABS(   TS ) .LT.   TOL ) GO TO 1                        HITS3204
         IF(      T .LT.  TC )  GO TO 4                                   HITS3205
         VO = VCO + DELV                                                  HITS3206
         CALL  TVX( VO  , TS , VX , XS )                                  HITS3207
         X = XS +XC                                                       HITS3208
      IF( IDUMP   .EQ. 1 ) WRITE(6,1003) T, XC,  VX , X , VCO , TCO       HITS3209
     1. A2X , CDAOB , CDAB , EF , EFO                                     HITS3210
          RETURN                                                          HITS3211
 1      CONTINUE                                                          HITS3212
C      T = TC                                                             HITS3213
         VX = VCO + DELV                                                  HITS3214
             X   = XC                                                     HITS3215
      IF( IDUMP   .EQ. 1 ) WRITE(6,1C03) T, XC,  VX , X , VCO , TCO       HITS3216
     1. A2X , CDAOB , CDAB , EF , EFO                                     HITS3217
        RETURN                                                            HITS3218
C    T LESS TC                                                            HITS3219
 4      CONTINUE                                                          HITS3220
          TGO = T                                                         HITS3221
        CALL TVX( VO , TGO ,   VXO , XG   )                               HITS3222
         DO  2  II = 1, 100                                               HITS3223
         A2X = A2( XG )                                                   HITS3224
        CDAOB=CDB + DLOG( (CDB +CAYD/VXO/VXO )/( CDB + CAYD/VO/VO ) )     HITS3225
     1 / 2.DO /F / XG                                                     HITS3226
        CDAB = CDAOB  +(CNA + CAA) / XG   * A2X                           HITS3227
         EF = DEXP( F  * CDAB * XG)                                       HITS3228
         EFO =DEXP( F  * CDAOB* XG)                                       HITS3229
 1003 FORMAT(     '       T            XC            VX            X     VHITS3230
     1CO          TCO          A2X         CDAOB         CDAB         EF    HITS3231
     2      EFO'/1X,1P11E12.4//)                                          HITS3232
         DELTT = ((EF -1.DO) / CDAB  - (EFO -1.DO )/ CDAOB)/F /VC         HITS3233
         DELV= VO * ( 1.DO/EF   - 1.DO/EFO )                              HITS3234
          TG = TGO   + DELTT                                              HITS3235
          VX = VXO   + DELV                                               HITS3236
       IF(IDUMP   .EQ. 1 ) WRITE(6,1005) TGO , XG, VXO , DELTT , DELV     HITS3237
     1 , VX   , TG  ,   VO                                                HITS3238
 1005 FORMAT(     '       TGO          XG            VXO          DELTT       DEHITS3239
     1LV          VX           TG          VO'/1X,1P11E12.4  // )         HITS3240
          IF( DABS( TG -T)  .LE. .00001 ) GO TO 3                         HITS3241
          XG = XG -VX * (TG -T )                                          HITS3242
         CALL   VXT(VXO, TGO , VO, WX, F , B , CDB ,XG, CAYD, TOL )       HITS3243
 2      CONTINUE                                                          HITS3244
          WRITE(6,1001)                                                   HITS3245
```

D.54 Subroutine TRAJT (concluded)

```
 1001   FORMAT(  '  TG DID NOT CONVERGE IN 100 TRIES.......' /// )         HITS3246
              RETURN                                                       HITS3247
 3      CONTINUE                                                           HITS3248
             X   = XG                                                      HITS3249
             RETURN                                                        HITS3250
               END                                                         HITS3251
```

D.55 Subroutine TRAJX

```
      SUBROUTINE TRAJX(  X , XC , VX  , T  )                           HITS3252
       IMPLICIT REAL * 8(A-H , O -Z )                                   HITS3253
 COMMON/OEC/OE(600)                                                     HITS3254
  EQUIVALENCE        (VO     ,OE( 101)), (VWIND  ,OE( 102)),            HITS3255
A(PHIWIN ,OE( 103)), (RHO    ,OE( 104)), (CD1    ,OE( 105)),            HITS3256
B(V1     ,OE( 106)), (CD2    ,OE( 107)), (V2     ,OE( 108)),            HITS3257
C(XN     ,OE( 109)), (CNA    ,OE( 110)), (SM     ,OE( 111)),            HITS3258
D(CMQ    ,OE( 112)), (CMPA   ,OE( 113)), (TRIM   ,OE( 114)),            HITS3259
E(PNITRI ,OE( 115)), (ANGO   ,OE( 116)), (PHIANG ,OE( 117)),            HITS3260
F(RATE   ,OE( 118)), (PHIRAT ,OE( 119)), (GAMO   ,OE( 120)),            HITS3261
G(PHIGAM ,OE( 121)), (A      ,OE( 122)), (ELL    ,OE( 123)),            HITS3262
H(D      ,OE( 124)), (W      ,OE( 125)), (AIX    ,OE( 126)),            HITS3263
I(AIY    ,OE( 127)), (P      ,OE( 128)), (ALPCON ,OE( 129)),            HITS3264
J(CAA    ,OE( 130))                                                     HITS3265
  EQUIVALENCE        (IDUMP  ,OE( 204))                                 HITS3266
  EQUIVALENCE        (ALPHA  ,OE( 400)), (ATOT   ,OE( 401)),            HITS3267
A(ALPTRM ,OE( 402)), (ALPHDO ,OE( 403)), (B      ,OE( 404)),            HITS3268
B(BETTRM ,OE( 405)), (BETAO  ,OE( 406)), (BETADO ,OE( 407)),            HITS3269
C(BETA   ,OE( 408)), (ALPMAX ,OE( 409)), (ALPMIN ,OE( 410)),            HITS3270
D(CAYD   ,OE( 411)), (CDB    ,OE( 412)), (CMA    ,OE( 413)),            HITS3271
E(CMTHTD ,OE( 414)), (CMTHTA ,OE( 415)), (CAY1   ,OE( 416)),            HITS3272
F(CAY2   ,OE( 417)), (CDA0B  ,OE( 418)), (CDAB   ,OE( 419)),            HITS3273
G(DELTT  ,OE( 420)), (DELV   ,OE( 421)), (DELT   ,OE( 422)),            HITS3274
H(DELX   ,OE( 423)), (DYDXO  ,OE( 424)), (DZDXO  ,OE( 425)),            HITS3275
I(DYDT   ,OE( 426)), (DZDT   ,OE( 427)), (DELW   ,OE( 428)),            HITS3276
J(DELLAM ,OE( 429)), (EYEP   ,OE( 430)), (EMP    ,OE( 431)),            HITS3277
K(EDLT   ,OE( 432)), (FDLT   ,OE( 433)), (F      ,OE( 434)),            HITS3278
L(H      ,OE( 435)), (PSIDO  ,OE( 436)), (PHIO   ,OE( 437)),            HITS3279
M(PHI1   ,OE( 438)), (PHI2   ,OE( 439)), (PSIO   ,OE( 440)),            HITS3280
N(R1     ,OE( 441)), (R2     ,OE( 442)), (R3     ,OE( 443)),            HITS3281
O(R4     ,OE( 444)), (RTRIM  ,OE( 445)), (TC     ,OE( 446)),            HITS3282
P(TCO    ,OE( 447)), (TS     ,OE( 448)), (TGO    ,OE( 449)),            HITS3283
Q(TP     ,OE( 450)), (THETAO ,OE( 451)), (THETDO ,OE( 452)),            HITS3284
R(TOL    ,OE( 453)), (VC     ,OE( 454)), (VCO    ,OE( 455)),            HITS3285
S(VYA    ,OE( 456)), (VZA    ,OE( 457)), (WX     ,OE( 458))             HITS3286
  EQUIVALENCE        (WZ     ,OE( 459)), (WO     ,OE( 460)),            HITS3287
A(W1     ,OE( 461)), (W2     ,OE( 462)), (WL2    ,OE( 463)),            HITS3288
B(XLAM   ,OE( 464)), (XLAM1  ,OE( 465)), (XLAM2  ,OE( 466)),            HITS3289
C(XNU1   ,OE( 467)), (XNU2   ,OE( 468)), (XG     ,OE( 469)),            HITS3290
D(XMU    ,OE( 470)), (XJAY   ,OE( 471)), (XJAZ   ,OE( 472)),            HITS3291
E(XJA    ,OE( 473)), (YA     ,OE( 474)), (ZA     ,OE( 475)),            HITS3292
F(ALPH   ,OE( 476))                                                     HITS3293
      IF( X .LE.  XC ) GO  TO 1                                         HITS3294
       X1  = XC                                                         HITS3295
       GO TO  2                                                         HITS3296
1      X1  =  X                                                         HITS3297
2     CONTINUE                                                          HITS3298
   IF(  X1 .EQ. 0.) X1=1.D-10                                           HITS3299
      CALL   VXT(VCO , TCO , VO  ,WX  , F,B, CDB, X1 , CAYD, TOL )      HITS3300
   IF( DABS( X1) .EQ. 1.D-10)  VCO=VO                                   HITS3301
```

D.55 Subroutine TRAJX (concluded)

```
          A2X = A2( X1)                                                 HITS3302
           FX1 =  F *  X1                                               HITS3303
      CDAOB  = CD8  + DLOG( (CD8 + CAYD/ VCO/VCO)/(CD8 +CAYD/VO/VO)HITS3304
     1 ) / 2.DO / F / X1                                                HITS3305
       CDAB = CDAOB  + (CNA + CAA)/X1 * A2X                             HITS3306
       EFCX  =    DEXP(  FX1 *  CDAB )                                  HITS3307
       EFCXO =    DEXP(  FX1 *  CDAOB )                                 HITS3308
       DELV =  VO * (1.DO/ EFCX   -1.DO/ EFCXO  )                       HITS3309
       DELTT = ((EFCX  -1.DO )/ CDAB - (EFCXO -1.DO)/CDAOB)/F /VO  HITS3310
         VC =  VCO  + DELV                                              HITS3311
         TC = TCO  + DELTT                                              HITS3312
           IF(  X .LE.  XC )  GO TO  3                                  HITS3313
          XS =  X -XC                                                   HITS3314
      CALL  VXT(VX ,  TS   .VC .WX, F, B .CD8, XS, CAYD, TOL )          HITS3315
           T=  TS  + TC                                                 HITS3316
         GO TO 4                                                        HITS3317
3     CONTINUE                                                          HITS3318
         VX = VC                                                        HITS3319
         T  = TC                                                        HITS3320
4      IF( IDUMP    .EQ. 1) WRITE(6,1000)  X , T , X1, VCO , TCO        HITS3321
     1 ,  A2X ,  CDAOB  , CDAB  , EFCX , EFCXO                          HITS3322
1000  FORMAT(   '      X            T            X1           VCO      THITS3323
     1CO           A2X          CDAOB        CDAB         EFCX        EFXCO  HITS3324
     2    ' / 1X , 1P10E12.4   //  )                                    HITS3325
       IF( IDUMP    .EQ. 1) WRITE(6,1001) DELV , DELTT, VC, TC .XS.VX  HITS3326
1001  FORMAT(   '     DELV         DELTT        VC           TC       XHITS3327
     1S           VX     ' /  1P10E12.4  // )                           HITS3328
         RETURN                                                         HITS3329
          END                                                           HITS3330
```

D.56 Subroutine TRAJXF

```
      SUBROUTINE TRAJXF(  X , XC , VX  , T  )                       HITS3331
        IMPLICIT REAL * 8(A-H , O -Z )                              HITS3332
 COMMON/DEC/DE(600)                                                 HITS3333
 EQUIVALENCE            (VO      ,DE(   1)), (VWIND  ,DE(   2)),    HITS3334
A(PHIWIN ,DE(   3)), (RHO     ,DE(   4)), (CD1    ,DE(   5)),      HITS3335
B(V1     ,DE(   6)), (CD2     ,DE(   7)), (V2     ,DE(   8)),      HITS3336
C(XN     ,DE(   9)), (CNA     ,DE(  10)), (SM     ,DE(  11)),      HITS3337
D(CMQ    ,DE(  12)), (CMPA    ,DE(  13)), (TRIM   ,DE(  14)),      HITS3338
E(PNITSI ,DE(  15)), (ANGO    ,DE(  16)), (PHIANG ,DE(  17)),      HITS3339
F(RATE   ,DE(  18)), (PHIRAT  ,DE(  19)), (GAMO   ,DE(  20)),      HITS3340
G(PHIGAM ,DE(  21)), (A       ,DE(  22)), (ELL    ,DE(  23)),      HITS3341
H(D      ,DE(  24)), (W       ,DE(  25)), (AIX    ,DE(  26)),      HITS3342
I(AIY    ,DE(  27)), (P       ,DE(  28)), (ALPCON ,DE(  29)),      HITS3343
J(CAA    ,DE(  30))                                                 HITS3344
 EQUIVALENCE            (IDUMP   ,DE( 204))                         HITS3345
 EQUIVALENCE            (ALPHA   ,DE( 400)), (ATOT   ,DE( 401)),    HITS3346
A(ALPTRM ,DE( 402)), (ALPHDO  ,DE( 403)), (B      ,DE( 404)),      HITS3347
B(BETTRM ,DE( 405)), (BETA    ,DE( 406)), (BETADO ,DE( 407)),      HITS3348
C(BETA   ,DE( 408)), (ALPMAX  ,DE( 409)), (ALPMIN ,DE( 410)),      HITS3349
D(CAYD   ,DE( 411)), (CD8     ,DE( 412)), (CMA    ,DE( 413)),      HITS3350
E(CMTHTD ,DE( 414)), (CMTHTA  ,DE( 415)), (CAY1   ,DE( 416)),      HITS3351
F(CAY2   ,DE( 417)), (CDAOB   ,DE( 418)), (CDAB   ,DE( 419)),      HITS3352
G(DELTT  ,DE( 420)), (DELV    ,DE( 421)), (DELT   ,DE( 422)),      HITS3353
H(DELX   ,DE( 423)), (DYDXO   ,DE( 424)), (DZDXO  ,DE( 425)),      HITS3354
I(DYDT   ,DE( 426)), (DZDTO   ,DE( 427)), (DELW   ,DE( 428)),      HITS3355
J(DELLAM ,DE( 429)), (EYEP    ,DE( 430)), (EMP    ,DE( 431)),      HITS3356
K(EDLT   ,DE( 432)), (EDLT    ,DE( 433)), (F      ,DE( 434)),      HITS3357
L(H      ,DE( 435)), (PSIDO   ,DE( 436)), (PHIO   ,DE( 437)),      HITS3358
M(PHI1   ,DE( 438)), (PHI2    ,DE( 439)), (PSIO   ,DE( 440)),      HITS3359
N(R1     ,DE( 441)), (R2      ,DE( 442)), (R3     ,DE( 443)),      HITS3360
O(R4     ,DE( 444)), (RTRIM   ,DE( 445)), (TC     ,DE( 446)),      HITS3361
P(TCO    ,DE( 447)), (TS      ,DE( 448)), (TGO    ,DE( 449)),      HITS3362
Q(TP     ,DE( 450)), (THETAO  ,DE( 451)), (THETOO ,DE( 452)),      HITS3363
R(TOL    ,DE( 453)), (VC      ,DE( 454)), (VCO    ,DE( 455)),      HITS3364
S(VYA    ,DE( 456)), (VZA     ,DE( 457)), (WX     ,DE( 458))       HITS3365
 EQUIVALENCE            (WZ      ,DE( 459)), (WO     ,DE( 460)),    HITS3366
A(W1     ,DE( 461)), (W2      ,DE( 462)), (WL2    ,DE( 463)),      HITS3367
B(XLAM   ,DE( 464)), (XLAM1   ,DE( 465)), (XLAM2  ,DE( 466)),      HITS3368
C(XNU1   ,DE( 467)), (XNU2    ,DE( 468)), (XG     ,DE( 469)),      HITS3369
D(XMU    ,DE( 470)), (XJAY    ,DE( 471)), (XJAZ   ,DE( 472)),      HITS3370
E(XJA    ,DE( 473)), (YA      ,DE( 474)), (ZA     ,DE( 475)),      HITS3371
F(ALPH   ,DE( 476))                                                 HITS3372
      IF( X .LE.  XC ) GO  TO 1                                     HITS3373
        XI = XC                                                     HITS3374
        GO TO  2                                                    HITS3375
1      XI =  X                                                      HITS3376
2     CONTINUE                                                      HITS3377
   IF(  XI .EQ. 0.) XI=1.D-10                                       HITS3378
     CALL   VXT(VCO , TCO , VO ,WX , F,B, CD8, XI , CAYD, TOL )     HITS3379
   IF( DABS( XI) .EQ. 1.D-10)   VCO=VO                              HITS3380
```

D.56 Subroutine TRAJXF (concluded)

```
          A2X = A2F( X1)                                                  HITS3381
           FX1 =  F *  X1                                                 HITS3382
       CDAOB   = CD8   + DLOG( (CD8 + CAYD/ VCO/VCO)/(CD8 +CAYD/V0/V0)HITS3383
   1 ) / 2.D0 / F / X1                                                    HITS3384
        CDAB = CDAOB   + (CNA + CAA)/X1  * A2X                            HITS3385
        EFCX  =   DEXP(  FX1 *  CDAB )                                    HITS3386
        EFCXO =   DEXP(  FX1 *  CDAOB )                                   HITS3387
        DELV =  V0 * (1.D0/ EFCX   -1.D0/ EFCXO  )                        HITS3388
        DELTT = ((EFCX  -1.D0 )/ CDAB - (EFCXO -1.D0)/CDAOB)/F /V0        HITS3389
          VC =  VCO  + DELV                                               HITS3390
          TC = TCO  + DELTT                                               HITS3391
            IF(  X .LE.  XC )  GO TO  3                                   HITS3392
           XS =  X -XC                                                    HITS3393
       CALL  VXT(VX ,  TS   ,VC ,WX, F, B ,CD8, XS, CAYD, TOL )           HITS3394
            T=  TS  + TC                                                  HITS3395
          GO TO 4                                                         HITS3396
    3   CONTINUE                                                          HITS3397
          VX = VC                                                         HITS3398
          T  = TC                                                         HITS3399
    4    IF( IDUMP    .EQ. 1) WRITE(6,1000)  X , T , X1, VCO , TCO        HITS3400
     1 ,  A2X ,  CDAOB  , CDAB  , EFCX , EFCXO                            HITS3401
 1000  FORMAT(   '     X           T            X1           VCO         THITS3402
     1CO         A2X        CDAOB        CDAB        EFCX        EFCXO    HITS3403
     2     ' / 1X , 1P10E12.4   //  )                                     HITS3404
        IF( IDUMP    .EQ. 1) WRITE(6,1001) DELV , DELTT, VC, TC ,XS,VX    HITS3405
 1001  FORMAT(   '     DELV          DELTT          VC           TC      XHITS3406
     1S         VX     ' /  1P10E12.4  // )                               HITS3407
          RETURN                                                          HITS3408
            END                                                           HITS3409
```

D.57 Subroutine TVX

```
      SUBROUTINE TVX(VI, T , VX , X )                                   HITS3410
          IMPLICIT REAL * 8(A-H , O -Z )                                HITS3411
      COMMON/DEC/DE(600)                                                HITS3412
      EQUIVALENCE           (IDUMP  ,DE( 204))                          HITS3413
      EQUIVALENCE           (ALPHA  ,DE( 400)), (ATOT   ,DE( 401)),     HITS3414
     A(ALPTRM ,DE( 402)), (ALPHDO ,DE( 403)), (B      ,DE( 404)),       HITS3415
     B(BETTRM ,DE( 405)), (BETAO  ,DE( 406)), (BETADO ,DE( 407)),       HITS3416
     C(BETA   ,DE( 408)), (ALPMAX ,DE( 409)), (ALPMIN ,DE( 410)),       HITS3417
     D(CAYD   ,DE( 411)), (CD8    ,DE( 412)), (CMA    ,DE( 413)),       HITS3418
     E(CMTHTO ,DE( 414)), (CMTHTA ,DE( 415)), (CAY1   ,DE( 416)),       HITS3419
     F(CAY2   ,DE( 417)), (CDAOB  ,DE( 418)), (CDAB   ,DE( 419)),       HITS3420
     G(DELTT  ,DE( 420)), (DELV   ,DE( 421)), (DELT   ,DE( 422)),       HITS3421
     H(DELX   ,DE( 423)), (DYDXO  ,DE( 424)), (DZDXO  ,DE( 425)),       HITS3422
     I(DYDT   ,DE( 426)), (DZDTO  ,DE( 427)), (DELW   ,DE( 428)),       HITS3423
     J(DELLAM ,DE( 429)), (EYEP   ,DE( 430)), (EMP    ,DE( 431)),       HITS3424
     K(EOLT   ,DE( 432)), (EDLT   ,DE( 433)), (F      ,DE( 434)),       HITS3425
     L(H      ,DE( 435)), (PSIDO  ,DE( 436)), (PHIO   ,DE( 437)),       HITS3426
     M(PHI1   ,DE( 438)), (PHI2   ,DE( 439)), (PSIO   ,DE( 440)),       HITS3427
     N(R1     ,DE( 441)), (R2     ,DE( 442)), (R3     ,DE( 443)),       HITS3428
     O(R4     ,DE( 444)), (RTRIM  ,DE( 445)), (TC     ,DE( 446)),       HITS3429
     P(TCO    ,DE( 447)), (TS     ,DE( 448)), (TGO    ,DE( 449)),       HITS3430
     Q(TP     ,DE( 450)), (THETAO ,DE( 451)), (THETDO ,DE( 452)),       HITS3431
     R(TOL    ,DE( 453)), (VC     ,DE( 454)), (VCO    ,DE( 455)),       HITS3432
     S(VYA    ,DE( 456)), (VZA    ,DE( 457)), (WX     ,DE( 458))        HITS3433
      EQUIVALENCE           (WZ     ,DE( 459)), (WO     ,DE( 460)),     HITS3434
     A(W1     ,DE( 461)), (W2     ,DE( 462)), (WL2    ,DE( 463)),       HITS3435
     B(XLAM   ,DE( 464)), (XLAM1  ,DE( 465)), (XLAM2  ,DE( 466)),       HITS3436
     C(XNU1   ,DE( 467)), (XNU2   ,DE( 468)), (XG     ,DE( 469)),       HITS3437
     D(XMU    ,DE( 470)), (XJAY   ,DE( 471)), (XJAZ   ,DE( 472)),       HITS3438
     E(XJA    ,DE( 473)), (YA     ,DE( 474)), (ZA     ,DE( 475)),       HITS3439
     F(ALPH   ,DE( 476))                                                HITS3440
        VWX = VI - WX                                                   HITS3441
        CAYBC = CAYD /B /CD8                                            HITS3442
        BCDK  = DSQRT( 1.D0 /CAYBC )                                    HITS3443
      TFCKT=DTAN(F* DSQRT(CD8 * CAYD) * T  )                            HITS3444
        VX =(VI  - WX  - TFCKT / BCDK) /( 1.D0 + BCDK * VWX* TFCKT) +WXHITS3445
        X =  WX * T  - DLOG( ( (VX-WX)** 2 + CAYBC)/( (VI-WX)**2 +CAYBCHITS3446
     1  )  )                                                            HITS3447
     2   /2.D0  /F / DSQRT(B)  /CD8                                     HITS3448
      IF(IDUMP.EQ.0) GO TO 69                                           HITS3449
        WRITE(6,1000) X, VX, CAYBC, VWX, BCDK, TFCKT                    HITS3450
   69 CONTINUE                                                          HITS3451
 1000   FORMAT(//' X=',  1P8E15.8)                                      HITS3452
           RETURN                                                       HITS3453
            END                                                         HITS3454
```

D.58 Subroutine T4NTV

```
      SUBROUTINE T4NTV ( IAW, IAY, LT, KBR )                              HITS3455
      IMPLICIT REAL*8 (A-H,O-Z)                                           HITS3456
      COMMON /CARDCM/  B(200), C(5000), IA(200,2), ID(10,2), INC,        HITS3457
     1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)    HITS3458
      SWF = 0.0D0                                                         HITS3459
      SYWF = 0.0D0                                                        HITS3460
      SYSWF = 0.0D0                                                       HITS3461
      DO 10 I=1,LT                                                        HITS3462
      K = IAW + I                                                         HITS3463
      SWF = SWF + C(K)                                                    HITS3464
      J = IAY + I                                                         HITS3465
      SYWF = SYWF + C(J) * C(K)                                           HITS3466
   10 SYSWF = SYSWF + C(J)**2 * C(K)                                      HITS3467
C NOMINAL VALUE                                                           HITS3468
      EY = SYWF / SWF                                                     HITS3469
      B(KBR+1) = EY                                                       HITS3470
C VARIANCE.                                                               HITS3471
      B(KBR+3) = SYSWF / SWF - EY**2                                      HITS3472
C TOLERANCE.                                                              HITS3473
      B(KBR+2) = 3.0D0 * DSQRT ( B(KBR+3) )                               HITS3474
      RETURN                                                              HITS3475
      END                                                                 HITS3476
```

D.59 Subroutine VXT

```
      SUBROUTINE  VXT(VX ,T , VI , WX, F , B, CD8 , X ,CAYD,TOL )
        IMPLICIT REAL * 8(A-H , O-Z )
  COMMON/OEC/OE(600)
  EQUIVALENCE        (IDUMP  ,OE( 204))
        F2 = 2.D0 * F
        VIWX = VI -WX
        VIWX2 =  VIWX * VIWX
          BCK = B *CD8 / CAYD
        BCD8KD  =  DSQRT( BCK )
        T1 = DATAN( BCD8KD * VIWX )
         TO = 0.D0
         T = 0.D0
         F2BC = F2 * DSQRT(B) * CD8
         VXO = 0.D0
          FSCDKD = F * DSQRT( CD8 * CAYD)
        DO  1     I  = 1,  100
          E =  FEXP( F2BC * (WX * T -X )      )
       VX =  DSQRT(VIWX2 * E  +( E-1.D0)/ BCK  )  + WX
       T  =           (         T1 -DATAN( BCD8KD*(VX-WX) )    )
     1  /  FSCDKD
  IF(IDU' EQ.0) GO TO 69
     1        6,1000 )I,T,TO , VX , VXO  , E
69  CONTI
1000 FORMAT   ( ',I3,'T =' , 1P1E18.8,' T-OLD=',1P1E18.8,'  VX =',1P1E18
     1.8 , '  VX-OLD=' ,1P1E18.8, '    E=',1P1E18.8 / )
        IF(  DABS( (VX-VXO  )/VX     ) + DABS((T-TO)/T ).LT. TOL)
     1  RETURN
         VXO = VX
         TO = T
 1       CONTINUE
        WRITE(6,1001)
1001    FORMAT(  '  VX AND T  ROUTINE  EXCEEDED 100 ITERATIONS...' //)
        RETURN
          END
```

D.60 Subroutine WIND

```
SUBROUTINE WIND( VX , T , X, WZ, WX,VO, VZ  ,  Z  )             HITS3511
     IMPLICIT REAL * 8(A-H, O -Z )                              HITS3512
  VOWX = VO -WX                                                 HITS3513
 VZ  =  WZ * (1.DO - (VX -WX )/ VOWX  )                         HITS3514
 Z =   WZ / VOWX * (VO * T - X )                                HITS3515
      RETURN                                                    HITS3516
        END                                                     HITS3517
```

D.61 Subroutine XXC

```
      SUBROUTINE XXC (XLO , XC , VO , DELL , CAY1 ,CAY2, TOL , AC )HITS3518
      IMPLICIT REAL * 8(A-H , O-Z )                                 HITS3519
      COMMON/OEC/OE(600)                                            HITS3520
      EQUIVALENCE        (IDUMP  ,OE( 204))                         HITS3521
        IF(  CAY1+ CAY2.NE. 0.)  GO TO 2                            HITS3522
         XC=1.D-3                                                   HITS3523
        RETURN                                                      HITS3524
 2     CONTINUE                                                     HITS3525
          XLV = XLO / VO                                            HITS3526
          XCO= DLOG(  AC /(CAY1 + CAY2 )   ) / XLV                  HITS3527
          DLVO=  DELL / VO                                          HITS3528
           DO  1   I = 1, 100                                       HITS3529
            E   = DEXP(    DLVO * XCO )                             HITS3530
            E1 =   CAY1 * E                                         HITS3531
            E2 =   CAY2 / E                                         HITS3532
           A   =  E1 + E2                                           HITS3533
           A2 =  E1 - E2                                            HITS3534
             B =   FEXP(- XCO   * XLV )                             HITS3535
         C    = (     A   - AC     * B)/( A * XLV   + DLVO * A2 )   HITS3536
            XC =   XCO -C                                           HITS3537
      IF(IDUMP.NE.1) GO TO 69                                       HITS3538
           WRITE(6,1000)  I , XC   ,   XCO ,   C   , E , E1, E2 , A, A2,B   HITS3539
 69   CONTINUE                                                      HITS3540
 1000  FORMAT(    '    NO.=', I4,' XC=' , 1P1E18.8, '  XC-OLD =', 1P1E18.8HITS3541
     1 , '  CHANGE = ' , 1P1E18.6  / 1P8E15.6    )                  HITS3542
          IF(   DABS( C/ XC ) .GT. TOL )GO TO 5                     HITS3543
        IF( XC  .LT.1.D-3) XC=1.D-3                                 HITS3544
         RETURN                                                     HITS3545
 5       CONTINUE                                                   HITS3546
            XCO = XC                                                HITS3547
   1   CONTINUE                                                     HITS3548
          WRITE(6,1001)                                             HITS3549
 1001 FORMAT(' XXC ROUTINE EXCEEDED 100 ITERATIONS WITHOUT CONVERGING'/)HITS3550
      IF(XC.LT.1.D-3) XC = 1.D-3                                    HITS3551
          RETURN                                                    HITS3552
           END                                                      HITS3553
```

D.62 Function ZATAN2

```
      FUNCTION ZATAN2(R1, R2)                                          HITS3554
         IMPLICIT REAL * 8(A-H , O -Z )                                HITS3555
      IF( R1    .NE. 0.)            GO TO 1                            HITS3556
      IF( R2    .NE. 0.)            GO TO             1                HITS3557
        ZATAN2=0.D0                                                    HITS3558
      RETURN                                                           HITS3559
    1   ZATAN2= DATAN2(R1, R2)                                         HITS3560
      RETURN                                                           HITS3561
      END                                                              HITS3562
```

D.63 Subroutine ZZ

```
      FUNCTION ZZ(CNA,CD8,CAYD,V0,EMP,ICNCL)                      HITS3563
      IMPLICIT REAL * 8 (A-H,O-Z)                                 HITS3564
      IF(ICNCL.EQ.1) GO TO 1                                      HITS3565
      ZZ = CNA * V0/EMP                                           HITS3566
      RETURN                                                      HITS3567
    1 CLA = CNA - (CD8 + CAYD/V0/V0)                              HITS3568
      ZZ = CLA * V0/EMP                                           HITS3569
      RETURN                                                      HITS3570
      END                                                         HITS3571
```

DISTRIBUTION

	Copies
A. Department of Defense	
Defense Advanced Research Projects Agency ATTN: Mr. Charles R. Lehner, Jr. 1400 Wilson Blvd. Arlington, VA 22209	1
Defense Documentation Center ATTN: TIPDR Cameron Station Alexandria, VA 22314	2
B. Department of the Army	
HQ DA (DAMA-WSZ, Dr. V. Garber) Washington, DC 20310	1
HQ DA (DAMA-WSW, LTC Gibson) Washington, DC 20310	1
Commander US Army Material Command ATTN: AMCRD-T, Dr. H. M. El-Bisi	1
AMCRD-T, Mr. D. Stefanye	1
AMCRD-FW 5001 Eisenhower Avenue Alexandria, VA 22333	1
Commander US Army Missile Command ATTN: AMSMI-RGT, Mr. Knoblach	1
AMSMI-RDF, Mr. Morrow Huntsville, AL 35801	1
Commander US Army Armament Command ATTN: AMSAR-RD, Mr. J. Brinkman	1
AMSAR-RDT, Mr. L. Johnson Rock Island, IL 61201	1
Commander US Army Training Doctrine Command ATTN: ATCD-CM-A Fort Monroe, VA 23651	1

DISTRIBUTION (Cont'd)

	Copies
Commander US Army Aviation Systems Command ATTN: AMSAV-EVW P. O. Box 209 St. Louis, MO 63166	1
Director US Army Ballistic Research Laboratories ATTN: AMXBR-CA, Dr. Eichelberger	1
Mr. C. M. Frank	3
Mr. A. S. Elder Aberdeen Proving Ground, MD 21005	1
Director AMXSY-ADE AMSAA ATTN: Mr. John L. Meredith Aberdeen Proving Ground, MD 21005	1
Commander US Army Air Defense Agency Fort Bliss, TX 79916	1
Commandant US Army Armor School Director, Combat Development ATTN: ATSD-CD-MM, Mr. M. Falkovich Fort Knox, KY 40121	1
Commandant US Army Air Defense School ATTN: ATSAD-DL-P Fort Bliss, TX 79916	1
President US Army Air Defense Board ATTN: STEBD-TD-S Fort Bliss, TX 79916	1
Commander Edgewood Arsenal ATTN: SAREA-CL-PA, Mr. A. Flatau Edgewood, MD 21010	1

DISTRIBUTION (Cont'd)

	Copies
Commander	
Frankford Arsenal	
ATTN: SARFA-J700, Mr. C. M. Dickey	3
SARFA-MDS-BP, Mr. P. Ayyoub	1
Mr. A. Korr, Chairman-Invention Evaluation Committee Building 64-4	1
Dr. Sid Ross	1
Philadelphia, PA 19137	
Commander	
Picatinny Arsenal	
ATTN: SARPA-FR-E, Mr. Hirskowitz	1
SARPA-AD-D-A-2, Mr. Michelson	1
SARPA-AD, Mr. Painter	1
Dover, NJ 07801	
Commander	
Rock Island Arsenal	
ATTN: SARRI-LA, Mr. Wrenn	1
SARRI-LE	1
SARRI-LA, Mr. Wells	1
SARRI-LW, Mr. Stickling	1
SARRI-LR-Y	15
Rock Island, IL 61201	
Commander	
Watervliet Arsenal	
ATTN: Mr. Austin	1
Mr. Rummel	1
Watervliet, NY 12189	

C. Department of the Air Force

AFAL/DLDG (Maj. Bilsbury)	
Eglin AFB, FL 32542	1
AFATC/DLM (Mr. Walter Detrick)	
Eglin AFB, FL 32542	1

DISTRIBUTION (Concl'd)

	Copies
AEDC/ARO (Mr. G. Norfleet) Arnold AFS, TN 37389	1

D. Department of the Navy

Commander Naval Surface Weapons Laboratory	
ATTN: Dr. V. Dawson	1
Mr. N. Sheetz	1
White Oak, MD 25989	
Commander Naval Research Laboratory	
ATTN: Mr. W. Atkins	1
Washington, DC 20315	
Naval Weapons Center Code 5115	
ATTN: Mr. Ward	1
China Lake, CA 93555	

E. General

Avco Systems Division	
ATTN: Dr. W. G. Reinecke	20
201 Lowell Street Wilmington, MA 01887	